MICROPROCESSOR BASED SYSTEM FOR THE DEVELOPMENT OF CONTROL AND PROTECTION OF HVDC CONVERTORS

A thesis
presented for the degree of
Doctor of Philosophy in Electrical Engineering
in the
University of Canterbury
Christchurch
New Zealand

by

J.C. Graham B.E. (Hons)

1984

TABLE OF CONTENTS

TABLE of FIGURES

Symbols and Abbreviations

AC	Alternating Current
A/D	Analogue to Digital
CF	Commutation Failure
Cf	Filter Capacitance
Ci	Commutating Voltage Zero Crossing i
Crossing	Commutating Voltage Zero Crossing
DA	Delay Angle
DAMIN	Minimum Delay Angle
DAo	Minimum Delay Angle Order
DC	Direct Current
Dcct	DC Current Transformer
E	Convertor Input Voltage
EAMIN	Minimum Extinction Angle
EAo	Minimum Extinction Angle Order
EA	Extinction Angle
FT	Fire Through
FP	Firing Pulse
HVDC	High Voltage Direct Current
K bytes	1024 bytes
Id	Direct Current
I/O	Input Output
Ls	System Inductance
MF	Misfire
Off	Valve turn off
On	Valve turn on
p.u.	Per Unit
Ram	Random Access Memory
Rom	Read Only Memory
SCR	Short Circuit Ratio
Vi	Valve i
Xc	Commutating Reactance
Xt	Transformer Leakage Reactance
α	Delay Angle
μ	Commutation Angle
γ	Extinction Angle

ABSTRACT

This project investigates aspects of microprocessor based control and protection schemes for high voltage direct current convertors.

To enable this investigation to be carried out, a multiple microprocessor HVDC development system has been assembled. This provides the necessary hardware resources, as well as providing the software development facilities necessary for the implementation of real time control and protection tasks.

In order to assess and optimise the performance of the various control and protection systems, considerable interactive monitoring facilities are provided as part of the HVDC development system's software.

The development system is used to implement real time control of a small scale convertor model. It is also used to implement a new form of convertor fault detection , which is in turn used as the basis for an implementation of convertor fault development control techniques for the first time on an operating convertor.

The operation of the fault development control scheme is examined in some detail, and results are presented showing its operation over a wide variety of fault types and system conditions.

ACKNOWLEDGEMENTS

I wish to express my deep gratitude to my supervisor Professor Arrillaga for his advice and assistance throughout the course of this research and in particular, for providing the financial support that enabled me to under take this research.

Thanks are due also like to thank my postgraduate colleagues for their assistance and encouragement, in particular to Dr Hiroshi Hisha, Dr Mike Parr, Alex Joosten and Steve McNeill.

I would also like to thank Charles Lake and Mike Dewe for their advice in the early stages of this project.

Grateful thanks are also due to the departments' technicians, in particular Mr A.R. Cox, for providing the technical support essential in a project of this kind.

Finally, I would like to thank my family and friends for their generous encouragement and support throughout my university studies.

1.0 Introduction.

HVDC techniques are finding a more common acceptance, with many large new schemes being commissioned or planned in recent years. As the size of the schemes are increased, so the complexity of the associated control and protection systems also increases. The current approach is to implement the control and protection systems in analogue hardware. This tends to produce a relatively inflexible system in which major changes or improvements are difficult to implement. It is possible, by using microprocessors to implement the control and protection systems, to introduce more flexibility with the use of appropriate software.

In recent years, microprocessors have been applied to the control or protection of a wide variety of processes. The task of controlling an HVDC convertor in real time is particularly demanding. The microprocessor must perform the necessary control calculations and, for a six pulse convertor, fire a new valve every 3.33mS while at the same time deriving any data from the convertor that it needs. Despite this, there have been several proposals for microprocessor (Reeve and Sevenco, 1978), or microcomputer, (Shore and Freris, 1978), based control schemes. The aim of this project has been to investigate possible applications of microprocessors to all aspects of HVDC transmission.

To aid this investigation a distributed processor system using four separate microprocessors has been designed and built to implement a microprocessor based HVDC Development System. The general form of this system is shown in figure 1.1.

This development system is used to implement a real time controller and convertor fault detection system. An application of the fault detection system to fault development control is examined in some detail. Neither the control or the fault detection system is intended to be a complete implementation of 'state of the art' techniques, but rather each has been implemented in a way suited to the microprocessor's operation, and each is examined to see what advantages or disadvantages accrue.

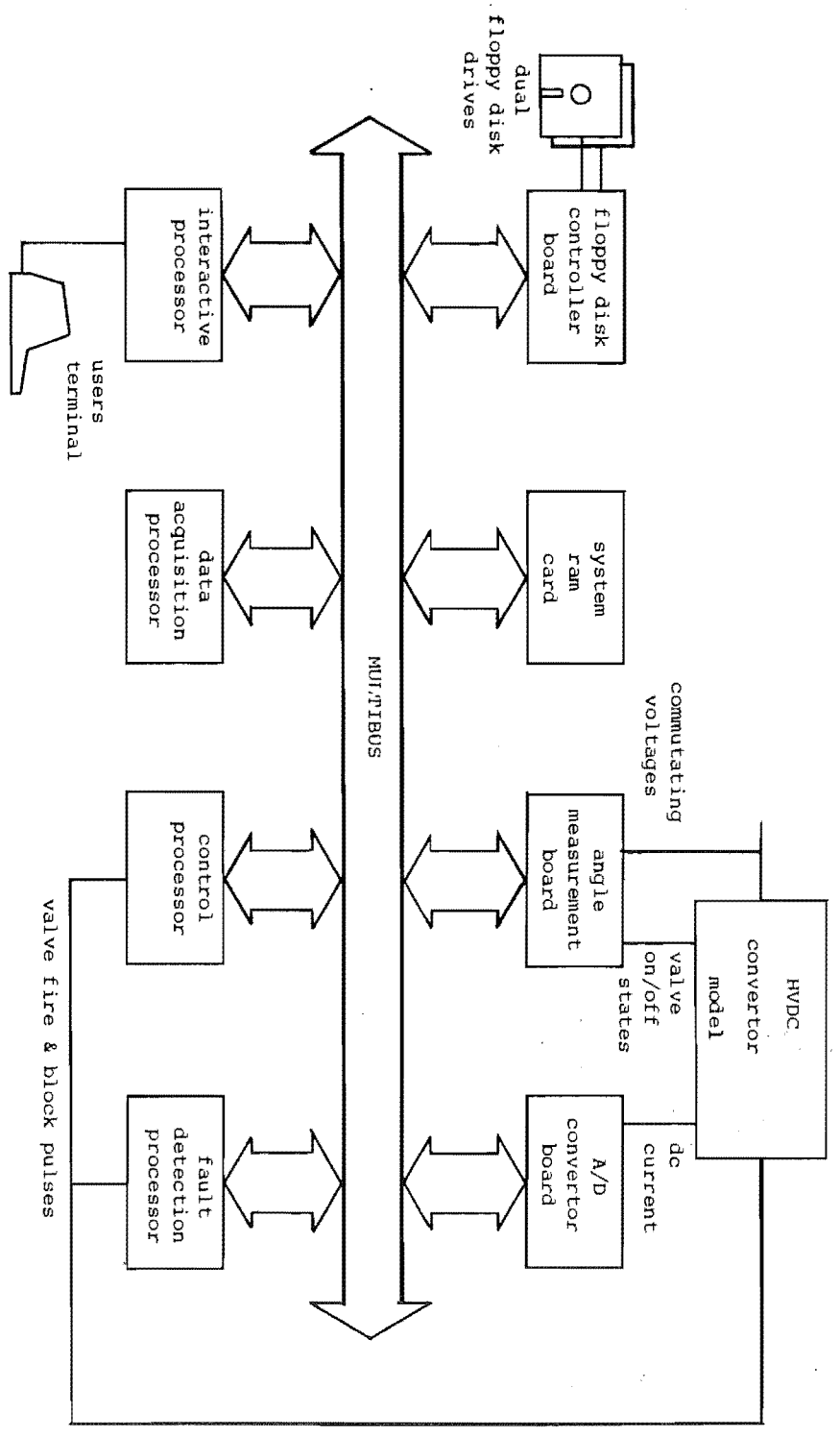


Figure 1.1 HVDC Development System Block Diagram

The basic structure of the hardware in the HVDC development system is presented in chapter 2. This details the hardware approach adopted, and explains why the structure of figure 1.1 resulted.

In order to make the system convenient to use, and to permit future use and expansion, complete self contained software development facilities have been provided. These are described in chapter 3.

Chapter 4 describes the data acquisition processes performed by the various sub-systems. A fast data acquisition system is needed both to provide data for the operation of the controller and fault detection as well as to provide sufficient information to analyse their operation and performance.

Chapter 5 deals with the real time control of the convertor. Existing control techniques are first reviewed, then early proposals for discrete logic gate based direct digital controllers are examined. Later microprocessor based systems are then reviewed before the control system implemented as part of this project is presented. Facilities for monitoring the performance of the microprocessor based controller are also described.

Chapter 6 begins by examining the likely convertor faults that may occur. A microprocessor based interactive fault simulation system is then developed. A novel form of information display is used to develop a convertor fault detection system capable of being implemented on a microprocessor. The operation of this fault detection system is then presented. Finally chapter 6 develops the principles of a fault development control system based on this fault detection scheme. The basic operation of the fault development control system is described, and some simple results explaining its operation on basic convertor faults are presented.

The operation of the fault development control system is examined in more detail in chapter 7. Results are presented that examine the operation of fault development control on weak ac systems under typical convertor and ac system faults.

Finally general conclusions are presented in chapter 8.

2.1 Introduction.

The hardware involved in the HVDC Development system consists of two basic items. The first is the actual HVDC converter model. This is the Hirollaga converter, developed by Dr. H. Hisha, and a short description will be included to familiarise readers with its capabilities (Hisha, 1983). The second item is the microprocessor system which constitutes the basis of this project and the bulk of this chapter will be given over to a description of it and an outline of why the approach adopted was chosen.

2.2 Convertor Model.

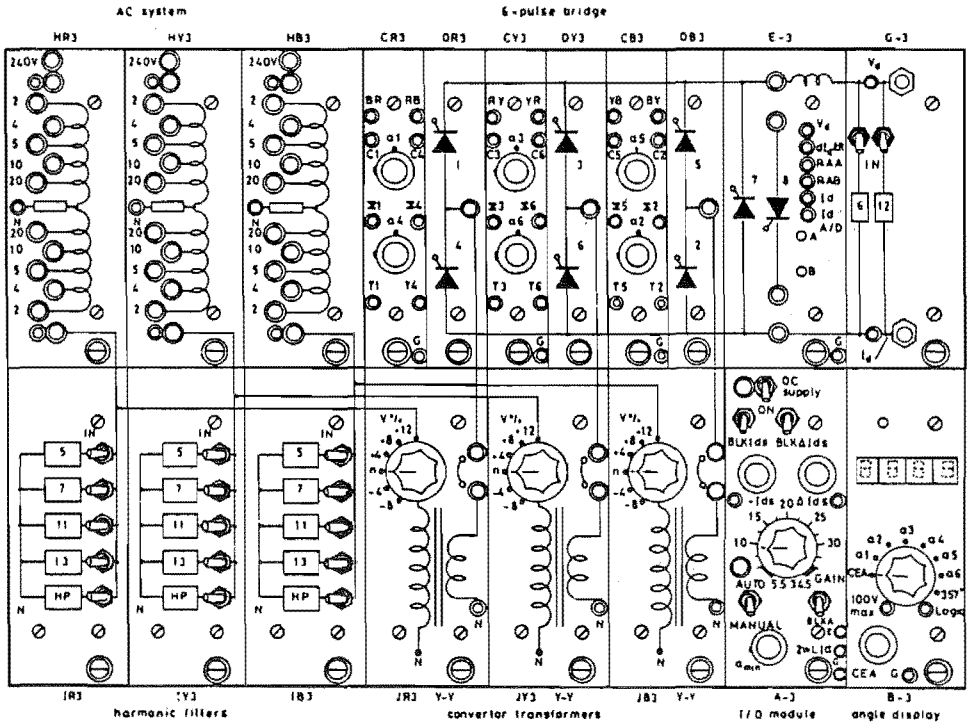


Figure 2.1 Convertor Front Panel Mimic

The central unit of the model is a 50 watt six-pulse bridge of modular design and its main constituent parts are illustrated in the front panel mimic diagram shown in fig 2.1 . The model has provision for both manual and automatic control, selected in module A3, which also contains the input and output signals used by the analogue predictive controller. The setting of the constant extinction angle controller is carried out in module B3 and the implementation of the automatic control in modules CR3-CY3-CB3 for red, yellow, and blue phases respectively. Alternative A, B, and C modules are available for an analogue based equidistant control scheme.

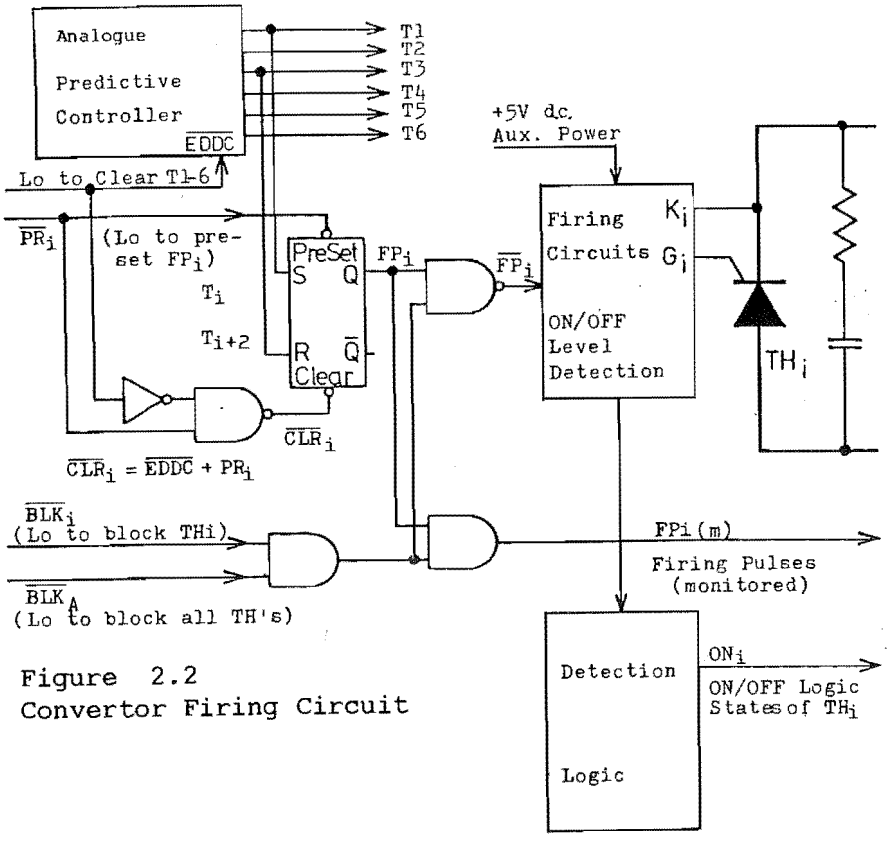


Figure 2.2
Convertor Firing Circuit

A block diagram of the valve firing circuits in the thyristor modules (module D) is shown in fig 2.2. Under predictive control the firing pulses FPi are produced in this module by means of a set-reset flip-flop driven by the Ti signals developed by the analogue controller. Blocking of individual valves or of the complete bridge can be achieved by the logic signals BLKi/ and BLKa/ respectively. Module D also produces the on/off states of the valves and their extinction angles EAi. Provision has been made for an external direct digital controller, under which the firing pulses are formed by a pair of logic signals PRi/ and CLRi/. External control is enabled by exerting the logic signal EDDC/ which disables the analogue trigger pulses.

The ac system impedance is simulated in module H by means of two variable reactors in series in each phase, with the central point connected to neutral through a shunt resistor which provides the desired impedance angle. The reactors are multiple tapped to provide a range of short circuit ratios (SCRs) between 2 and 20, which covers the range found in most practical schemes. This representation is only valid at the lower frequencies but is considered acceptable (Bowles, 1970) in the presence of ac harmonic filters.

Each phase of the model contains independently switchable filters for the 5th, 7th, 11th, and 13th harmonics as well as a high pass unit for higher order harmonics. These filters are of the traditional type, with a single tuned arm for each of the lower characteristic harmonics.

Two sets of transformers (module J) are available to model the star-star and star-delta configurations and each transformer has several taps which can be used to unbalance the supply voltage. Care was taken with the design of these transformers to ensure that they have realistic leakage reactances, and one set was designed with a value of 5%, while the second set has a value of 8%.

The convertor can be connected to a similar unit through a transmission line model also of modular construction, with provision to model varying lengths of both overhead line and underground cable.

2.3 Microprocessor System.

2.3.1 Introduction.

The aim of the project is to produce a microprocessor based HVDC development system to investigate possible applications of microprocessors to all aspects of HVDC transmission. The system would be required to provide full interactive facilities for program development, real time control operation, fault simulation, fault detection, and fault development control, and to provide comprehensive monitoring facilities to measure the performance of these sub-systems. It was also considered desirable that the system, where possible, be constructed from standard 'building blocks', and contain a minimal amount of dedicated specialised hardware.

With these requirements in mind it was first necessary to decide what type of microprocessor to use. The two main alternatives considered were whether to use a single very fast bit slice microprocessor or to use multiple general purpose processors and split the tasks between the individual processors. The following sections will deal initially with the factors influencing the final choice of processor and will then go on to describe the general structure of the overall microprocessor system.

2.3.2 Choice of Microprocessor.

2.3.2.1 Single or Multiple Processors.

A brief study of the requirements quickly indicated that if a single microprocessor were to be used, it would have to be extremely fast. The alternative was to use slower general purpose processors and split the tasks into groups that could each be handled by an individual processor in the time available.

The use of a single fast processor was quickly rejected for a number of reasons, the major reason being that the processor would, on occasions, have to perform some tasks simultaneously. This would be incompatible with the inherent serial nature of a microprocessor. The general unavailability of peripheral support components and the lack of high level programming languages also influenced the decision against the use of a single fast processor.

Therefore the approach adopted was to split the tasks into groups with no simultaneous timing requirements and allocate one processor for each group. A preliminary allocation was made with one processor to handle the operating system and interactive communications, one processor to handle the data acquisition, one processor to handle the controller and one processor to handle fault detection.

2.3.2.2 Processor Size.

The next major consideration in the choice of processor was to decide whether to use an eight bit microprocessor or one of the newer more powerful, and more expensive, sixteen bit processors.

Initially a prototype system was built around two Intel iSBC 80/30 (Intel Manual e) single board computers which provided much valuable information.

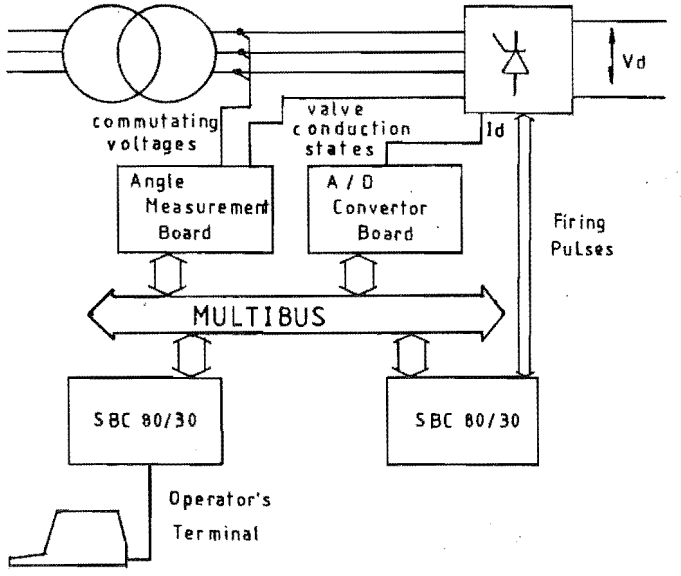


Figure 2.3 Prototype System

The basic structure of the test system, shown in fig 2.3 , consisted of two processor boards, each with an 8085 microprocessor and 16 Kbyte of memory, and a prototype set of data acquisition boards, the latter implemented the hardware parts of the convertor angle measuring system and also provided six eight bit analogue to digital convertor channels.

This system was used to test the basic data acquisition algorithms and was also used to implement a prototype control system in order to ascertain the feasibility of a microprocessor based controller.

The major conclusion reached with the preliminary system was that a microprocessor implementation was feasible for the controller, although the 8085 was not a suitable processor to use. The problem is that the 8085 has only a very limited sixteen bit instruction set and has no divide or multiply instructions of any kind. As most of the data to be processed consists of sixteen bit quantities, such as the timer values and the delay, commutation and extinction angles, and since it is necessary to perform multiplication as part of the control algorithm it was decided to use a suitable sixteen bit processor in the final HVDC development system. Possible processors included the Motorola 68000 family, the Intel iAPX86 family or the Zilog Z8000 family.

2.3.2.3 Departmental Support.

The final factor influencing the choice of processor was the facilities already available in the Department of Electrical and Electronic Engineering for both hardware and software development support.

For various reasons, primarily due to the different levels of product support in New Zealand available from the various manufacturers, the department uses the Intel family of microprocessor components, and has two Intel Series II Microprocessor Development Systems (Intel Manual d). These provide software support for the 8085, 8048, 8051 and 8086 families of microprocessors, as well as hardware support in the form of In Circuit Emulators (Intel Manuals g,h) for the 8085, 8048 and 8086. Cross software for the 8085 and 8086 processors was also available on the department's VAX 11/750 computer.

In view of the preference for a sixteen bit processor outlined above, it was decided to use the 8086 as the basic processing element.

2.3.2.4 Board Level Products.

For a microprocessor to be of any practical use, it must first be surrounded by various support components. To run a program it requires some memory, either ram or rom and to interface with the real world it usually requires some peripheral chips. To maintain the general nature of the processing elements it was decided that each microprocessor would be placed on a separate board with the necessary support components.

For the HVDC system it was envisaged that each processor would require at least one interrupt controller, several programmable timers, parallel I/O ports and probably a serial I/O port.

The memory requirements would be primarily determined by the anticipated size of the HVDC software. It was expected that the programs for the slave processors would be reasonably compact, but that the interactive routines would be relatively large. In addition it was known that the operating system under consideration would require up to 12K byte of rom and 140K byte of ram.

The other major requirement was that the ram on each processor be dual ported. That is that the ram be accessible to an external processor via some form of common bus as well as being accessible to the on-board processor. This was necessary for two major reasons. The first was that since all HVDC programs were of a developmental nature and continually liable to change, a 'final' version able to run in rom would never exist, and consequently the code must execute in ram. This then requires that the code be reloaded from disk as and when required. This in turn requires that the disk controller be able to access each processor's on-board ram. The other major reason for requiring dual ported memory was that it permitted one processor to access data prepared by another, reducing overheads and simplifying the establishment of a common data base for all processors.

2.3.2.5 Bus Structure.

It was considered desirable to have all processors operating on a common bus as this would permit the sharing of common resources such as global memory, dedicated data acquisition hardware and a disk controller. This requires that some form of bus access system be provided since each processor would require the use of the bus to communicate with the peripheral boards, and only one processor can use the bus at any one time without conflicts arising.

The department was already using the Intel bus system known as Multibus (Intel Manual m) . This has the facility to provide the arbitration between the various processors, and Intel supplies a bus arbiter chip, the 8289 that interfaces the 8086 processor to the Multibus.

The Multibus permits up to three masters using a simple serial daisy-chain priority resolution scheme in which a processor may obtain access to the bus only if no processor higher in the chain is using it. If a slightly more complicated parallel priority resolution scheme is used then up to sixteen processors can be conveniently connected. This is the method that has been adopted for the HVDC development system, and it operates by providing each processor board with a bus request and a bus granted line. All request lines are applied to a priority encoding circuit which signals the highest priority processor currently requesting use of the bus to a decoder circuit which activates the appropriate grant line. The actual exchange of the bus between two processors is handled entirely by the 8289 bus arbitration chip.

The Multibus is capable of supporting both eight and sixteen bit data transfers and has a standard address space of sixteen megabytes.

2.3.2.6 Local Boards.

An 8086 based board fulfiling most of the requirements outlined above was available in the department. These had originally been produced for a multiprocessor transient stability project and were now available for use. Unfortunately they had never been actually completely assembled, as the transient stability project did not require the full capabilities of

each board, and during assembly and subsequent testing several previously undetected design faults appeared. Considerable time was spent trying to fix these faults, ultimately with little success. In view of the poor results being obtained, it was decided to abandon them and purchase the necessary boards from commercial suppliers.

2.3.2.7 Commercial Boards.

A survey of possible boards indicated several that would fit the requirements, the final influencing factor being the use of Intel iSBC 86/12a (Intel Manual a) boards by other projects in the department. This board satisfied the memory requirements for the slave processors, containing up to 16K bytes of rom and 32K bytes of dual port dynamic ram.It also provided the I/O functions needed with an eight level interrupt controller, three sixteen bit programmable timers, three eight bit parallel I/O ports and a programmable serial port. A total of three Intel iSBC 86/12a's were purchased for use as the slave processor boards. In order to meet the larger memory requirements of the interactive board, a Matrox MBC 86-12-128 (Matrox Manual) was obtained. This is functionally equivalent to the Intel iSBC 86/12a, but contains room for up to 64K bytes of rom and 128K bytes of ram, at least partially meeting the memory requirements of the operating system.

2.3.2.8 Newer Processors.

The number and variety of microprocessors available seems to be increasing at an alarming rate, and the designer seems to be condemned to starting a project with a processor that will probably be obsolete by the time the project is completed. This project has proved to be no exception.

Since the project was started Intel has released the 80186, an upgraded 8086 with counter timers, interrupt controller, and indeed most of the i/o support found on the SBC 86/12a all combined into a single integrated circuit. This effectively makes most of the hardware on the SBC 86/12a redundant. Alternatively, if the processing needs to be split further then the 8096 single chip microprocessor is now available. This contains a sixteen bit cpu, counter-timers, interrupt controller, ten bit analogue to digital converter and 4K byte of eprom on a single chip, and could form the heart of a system with one processor per valve.

2.3.3 Support boards.

These boards provide the system with global resources accessible by all processors as opposed to the local resources that are located on each processor board. The support boards provided in the HVDC development system consist of two memory boards, one floppy disk controller board, and two data acquisition boards. The data acquisition boards comprise an analogue to digital converter board and an angle measurement board. The functions of these boards will be discussed in the following sections. Again, as was the case with the processor cards, general purpose commercial boards were used where possible in order to simplify design.

2.3.3.1 Memory boards.

It was considered imperative that no "instruction fetches" be performed through Multibus while the HVDC development system is running, as this would otherwise impose a prohibitively high bus usage, with a corresponding decrease in system speed. The 32K bytes of ram provided on each processor board available for the HVDC development system's use was considered adequate for the anticipated program size.

The monitoring system however requires a large amount of memory for data storage and so a 128K byte ram board was provided for this purpose. This was a board designed in the department, originally for the transient stability analysis project mention above. It is similar in operation to a number of commercial equivalents and provides 128K bytes of ram using 16K bit chips and also provides up to 16K bytes of eprom space. In addition to the monitoring system's storage space, the global data base was also placed on this board as it then made the database available to all processors at the same physical address.

The other major memory requirement was that needed by the operating system. The operating system chosen (chapter 3.1) requires 140K bytes of ram, with the linker and compilers requiring a further 90K bytes. The interactive processor card has a total of 128K bytes available of which 32k bytes have been reserved for the HVDC interactive software to run in. The remaining 96K bytes is used by the operating system, and a further 256K byte ram board is provided to complete the operating system address space.

This is a commercial board, an Intel SBC 056a (Intel Manual b). The use of this external memory board results in part of the operating system and all utility programs residing in off-board memory. The consequent drop in speed for the operating system is not considered to be important as, unlike the HVDC system, it is not required to respond to real time events.

Fig 2.4 shows the memory map of the overall system. Note that because of the dual-ported nature of the memory, the slave processor memory that appears on Multibus in the address ranges 60000 - 67FFF, 68000 - 6FFFF,and 70000 - 77FFF also appears in the range 0 - 07FFF to the particular on-board processor.

2.3.3.2 Disk Controller.

As the system was required to be capable of stand alone operation, some form of bulk storage was required for program source code as well as operating system utilities. The main choice was between using small capacity floppy disks and using a large capacity hard disk. The high cost of the latter meant that floppy disks appeared to be the most suitable form of mass storage, and consequently provision was made in the cabinet as shown in fig 2.5 for two Shugart SA800 (Shugart Manual) floppy disk drives. These are a single sided eight inch drive with a storage of approximately 640K bytes. The choice of disk controller is dependent on the operating system used, and for this reason the Intel iSBC208 floppy disk controller board (Intel Manual c) was selected. This is capable of controlling up to four 8" or four 5 1/4" drives.

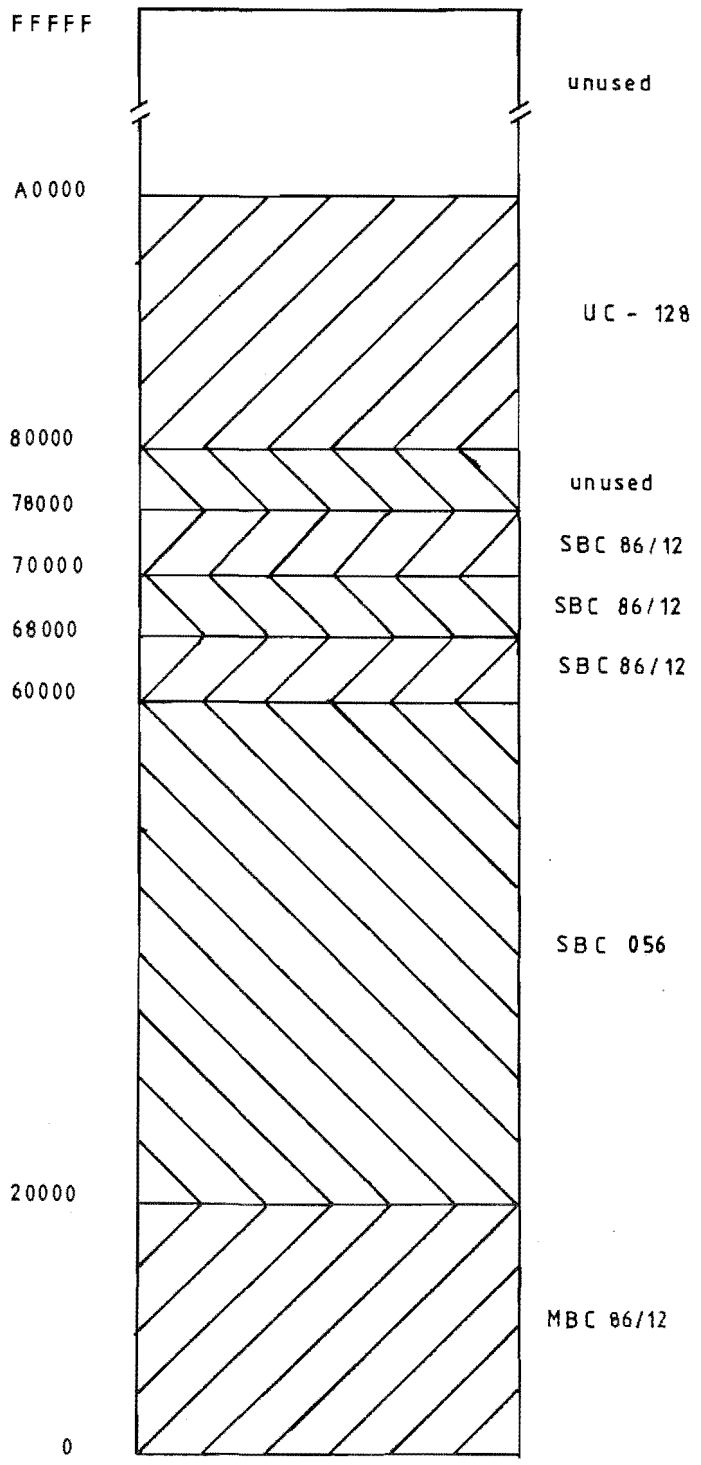


Figure 2.4 System Memory Map

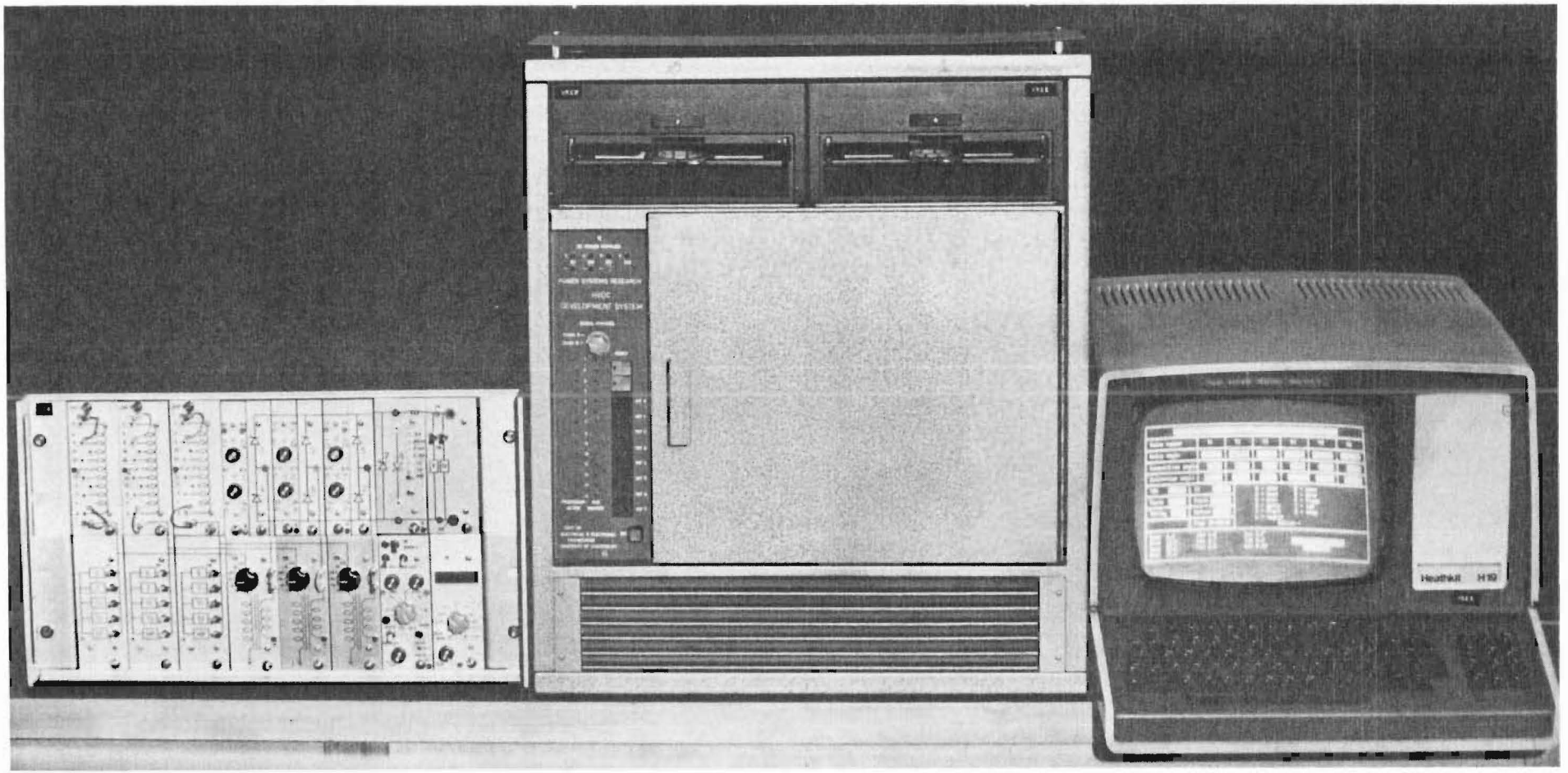


Figure 2.5 HVDC Development System

2.3.3.3 Analogue to Digital Convertor Board.

An a/d convertor board was built to meet the requirements that will be outlined in chapter 4.1.2. The major reason for building the board rather than using a commercially available one was that the multi-channel 12 bit a/d convertor boards available in New Zealand contain only one a/d convertor, and use an analogue multiplexer to select the channel to be converted. In this project however, it appeared to be desirable to be able to make several measurements simultaneously, for example dc voltage and current, or all 3 commutating voltage waveforms. Thus a board with eight individual a/d convertors was designed.

The actual a/d convertor chip used was the Analogue Devices AD574K (Analog Devices) . This is a twelve bit microprocessor compatible convertor with a 30 μs conversion time. Twelve bit a/d convertors were used for reasons explained in chapter 4 after experience with eight bit a/d convertors in the prototype system. Fig 2.6 shows the block diagram of the board.

As shown , sufficient board space was available for eight a/d convertors and their associated sample and hold circuits. The eight a/d convertors were arranged as two groups of four, with the operation of each group being independent of the other, thus permitting different sampling rates to be used on each group. The a/d convertor board occupies 32 locations in the i/o address space of the microprocessors.

The conversion process is initiated by writing to the sample/hold port of the required group to latch the sample and hold circuit in the hold state. Actual conversion is then started by writing to the appropriate start conversion port. This starts all four convertors in a group simultaneously. On completion of conversion by all four convertors an interrupt is generated to signal the controlling microprocessor that conversion is complete. This means that the controlling processor is free to run other tasks while the a/d convertor is actually performing the conversion. On completion the data is then available at the individual data ports.

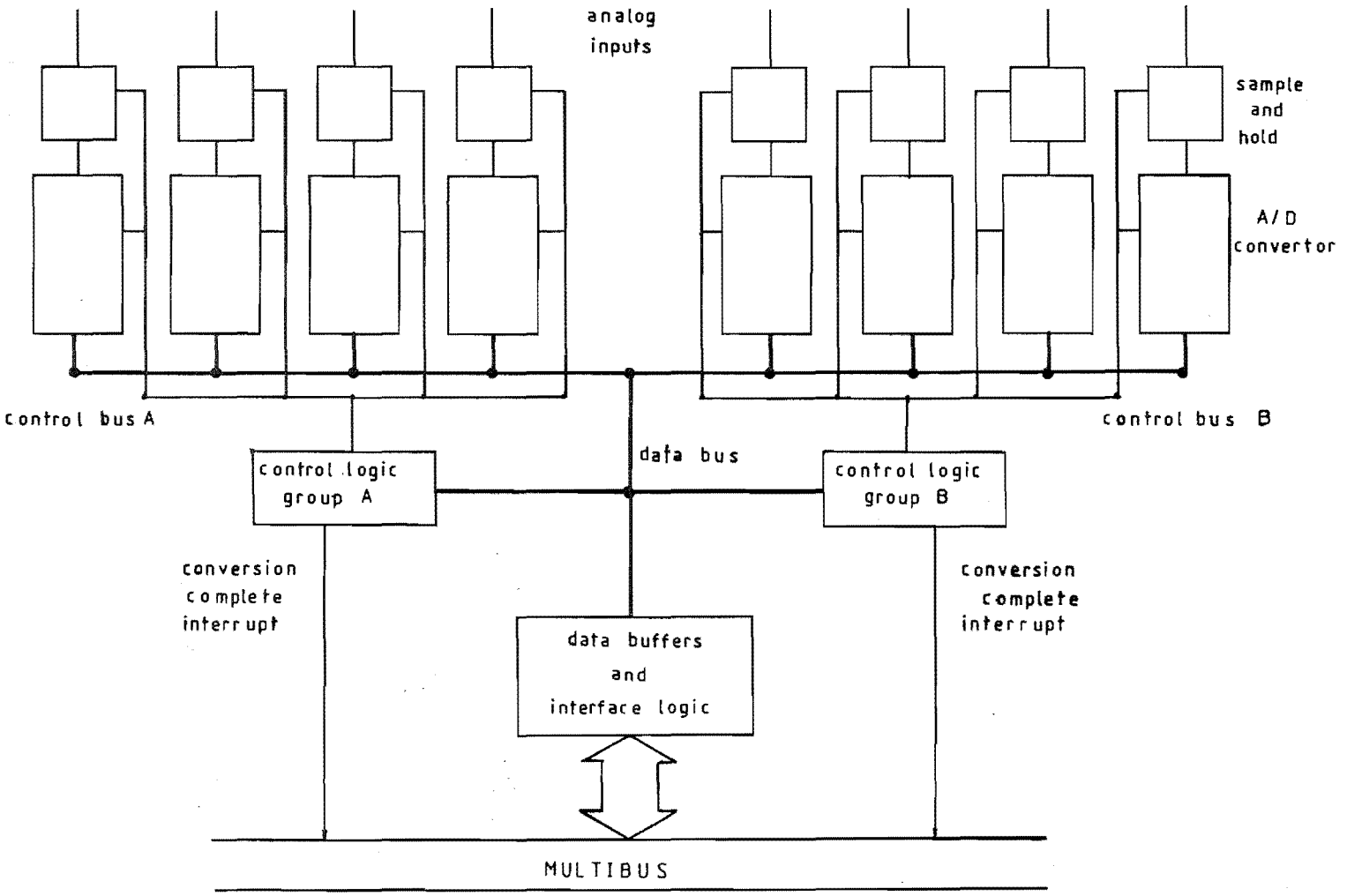


Figure 2.6 A/D Convertor Board Block Diagram

2.3.3.4 Angle Measurement Board.

The angle measurement board, shown in fig 2.7, contains a variety of hardware associated with the angle measurement process as well as other data acquisition hardware needed by the system. The actual operation of the angle measurement process will be described in the data acquisition chapter. This is the only board in the HVDC development system that is not of a general purpose nature, its design and operation being tailored to the specific application.

The first function the board provides is to buffer the signals from the HVDC convertor model. Fig 2.8 shows the buffer stages is detail.

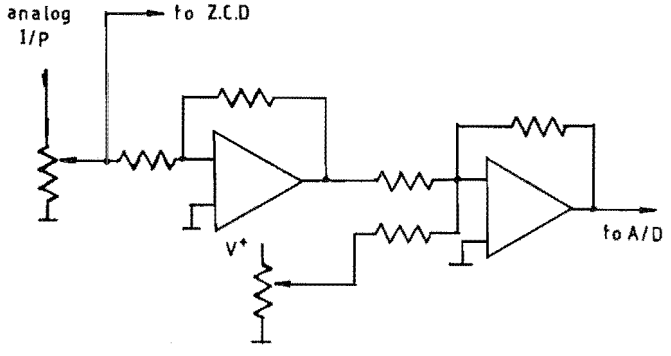


Figure 2.8 Buffer Stage

The commutating voltages and dc voltage and current signals are first buffered and then attenuated before being level shifted and then applied to the a/d convertor board. The incoming valve on/off state logic level signals are also buffered through schmidt triggers to improve the noise immunity.

The commutating voltage waveforms are also supplied to a set of zero crossing detectors located on the angle measurement board. Initially the zero crossing detectors merely 'squared up' the ac waveform. This technique however caused several problems. The first was that a small dc offset would cause one half cycle to appear to be longer that the other, and this adversely affected the overall accuracy of the angle measuring process. The second problem that occurred was that the notches in the commutating voltage waveforms would, at times, cause multiple crossings which also seriously degraded the measurement accuracy.

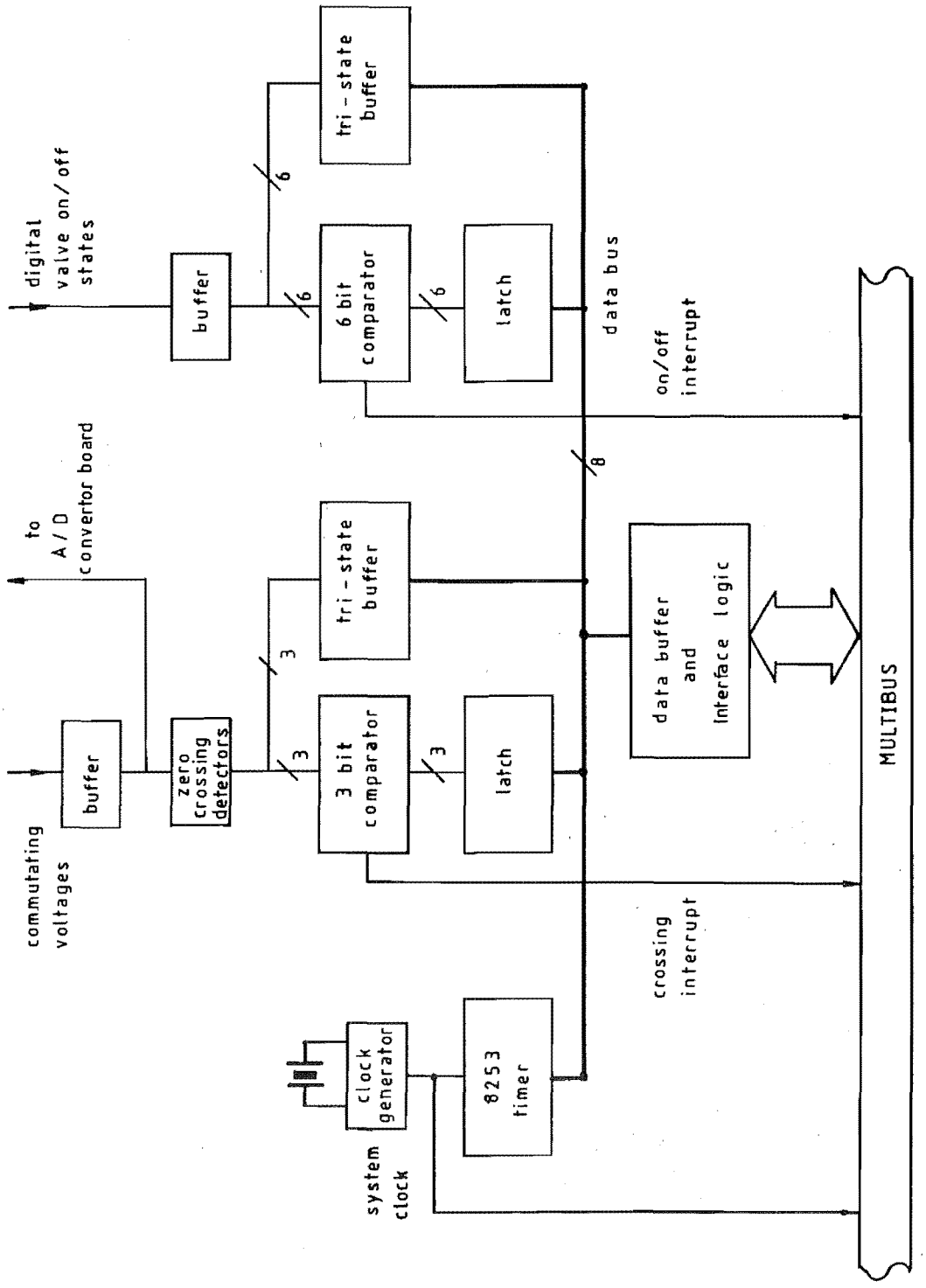


Figure 2.7 Angle Measurement Board Block Diagram

To overcome these problems a revised zero crossing circuit was implemented, and this is shown in fig 2.9 .

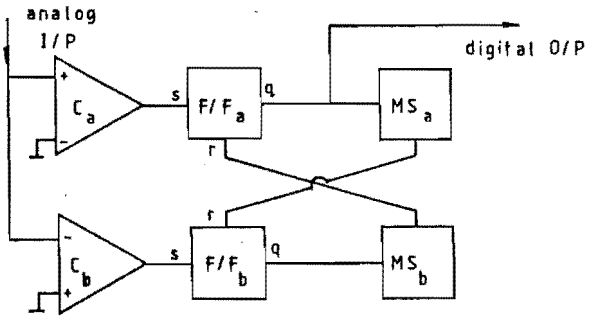


Figure 2.9 Zero Crossing Detector

Two comparators are used to allow both the positive and negative going crossing to be detected individually. The outputs of the comparators are used to toggle the flip flops F/Fa and F/Fb to the set state, while the monostables MSa and MSb reset the flip flops as required. As shown the positive and negative halves are cross coupled so that the positive going transition is detected by comparator A and F/Fa is set. At the same time the setting of F/Fa triggers MSa which resets F/Fb and holds it in the reset state for approximately 170 degrees.

Thus if a multiple crossing occurs only the first positive going crossing is captured and subsequent crossings are ignored for the next 170 degrees. When the first negative going crossing, after the reset pulse is removed, is detected then F/Fb is set and a reset pulse applied to F/Fa, again for about 170 degrees. Therefore only the first crossing in the right direction is detected, and this must occur in a narrow window near the true crossing. If a commutation notch does occur in this window then the effective zero crossing detection point will shift, as shown in fig 2.10 , but only one crossing will be recorded.

This means that a small error can sometimes occur, but normally the commutation notches are well away from the expected zero crossings and so are totally rejected.

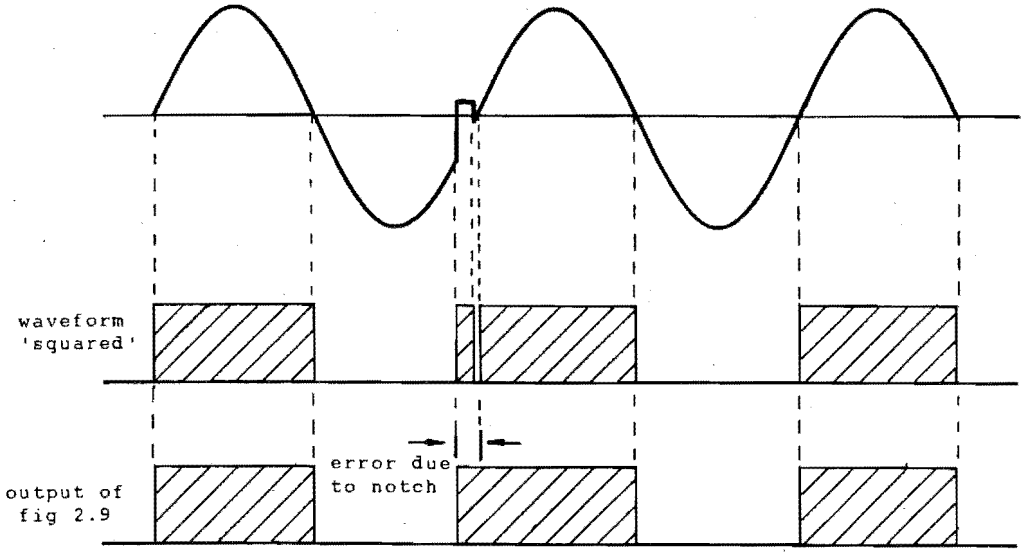


Figure 2.10 Zero Crossing Detector Waveforms

As described in chapter 4.3 the angle measuring hardware takes the form of two digital comparators which produce an interrupt whenever a transition of the valve on/off states or commutating voltages is detected. Associated with the two comparators is the 1.8 MHz system clock which provides the time of interrupt information, and is also supplied to the slave processor boards for other timing purposes. A 1.8 MHz clock was used as the system clock because it provided a basic timing interval of 0.01 degrees for a 50 Hz ac system. Fig 2.8 also shows the implementation of the system clock and digital comparators. The system clock has been implemented with an 8224 clock generator chip and an 8253 programmable counter timer chip. The digital comparators are implemented with 8131 bus comparator chips and the associated latches and tri-state buffers are standard TTL components. The exact mode of operation of the comparator and how it is used in the angle measurement process will be described in chapter 4.3.2 .

2.3.4 Card Cage and Power Supplies.

The multibus card cages used as the backplane were the AMC 95/6440. These have sockets for six multibus boards and two units are connected together to provide a total of twelve slots. Parallel priority resolution is provided for the top eight slots, the bottom four slots being occupied by the memory and data acquisition cards which need no resolution.

The power supply requirements for each board are tabulated below, with all currents in amps.

Supplies - Boards.	+5v	-5v	+12v	-12v	+15v	-15v	+24v
SBC 86/12a	5.5	-	0.35	0.04	-	-	-
MBC 86/12-128	5.0	-	0.025	0.09	-	-	-
SBC 208	3.0	-	-	-	-	-	-
SBC 056	5.0	-	-	-	-	-	-
UC 128K	2.7	0.006	0.1	-	-	-	-
A/D board	1.2	-	-	-	0.09	0.16	-
Angle meas	2.0	-	0.15	0.14	-	-	-
Disk drives	1.6	0.1	-	-	-	-	2.6
Totals.	37.0	0.106	1.325	0.27	0.09	0.16	2.6

Commercial switch mode power supplies were used for all except the minus five volt supply in order to minimise the space required and the heat produced inside the cabinet. The minus five volt supply is regulated from the minus twelve volt supply by a three terminal regulator. The total supply rating available is shown below.

Supplies -	+5v	-5v	+12v	-12v	+15v	-15v	+24v
Available -	46.0	1.5	6.3	6.3	1.0	1.0	3.2

Cooling for the supplies is provided by a total of three fans. In addition each of the card cages has a separate fan to ensure that sufficient air is kept moving across the surface of the boards. Finally the cabinet was designed so that the air exhausts over the disk drives, cooling them as well.

2.3.5 Terminal.

The standard terminal used in the department, the Zenith Z19, is also used in this project. This is a VT52 equivalent with a very limited graphics capability. The availability of a better terminal would enhance the presentation of some displays.

2.4 Conclusion.

The microprocessor system described in this chapter is contained in a single cabinet, shown in fig 2.5 together with the convertor model and the operators terminal. In conjunction with the operating system to be described in chapter 3 it provides a development system capable of being used to investigate the applications of microprocessors to most aspects of HVDC control and protection. It is also capable of being used as a stand alone microcomputer, and could, with minor changes to the data acquisition boards, be used to control many other processes.

3.0 Operating System and Software Support Facilities.

3.1 Operating System.

The requirement that the HVDC development system be capable of stand-alone operation requires that some form of operating system must be provided. This is needed to facilitate program development during the project and to permit software stored on floppy disks to be loaded into the microprocessors and executed. The exact choice of operating system was influenced by many factors, among them the languages supported, vendor supplied utilities, departmental support and the actual size of the operating system.

3.1.1 Choice of Operating System.

The basic requirement for the operating system was that it be capable of running on an 8086 microprocessor, be able to operate without needing a hard disk and be capable of providing the program development facilities required. This ruled out the use of a UNIX based system as these systems are very large and almost always require a large capacity hard disk to operate. The main choices appeared to be CP/M-86 from Digital Research , MS-DOS from Microsoft or RMX 86 from Intel .

CP/M-86 is a sixteen bit upgrade of the well known CP/M-80 operating system. Unfortunately there was relatively little information on it available here in New Zealand at the time the selection was being made, but from what information was available it appeared that it did not have compilers for a suitable high level language, and would not be easily interfaced to the HVDC programs.

MS-DOS is the commercial version of the IBM PC's operating system PC-DOS. As such it implied that a wide range of software languages and utilities would probably be available, but again there was little information available on it in New Zealand.

Since the decision on the choice of operating system was made more information on both the above operating systems has become available, and the objections to their use at the time are no longer valid. Either could now be used, and both now seem to provide the required level of software development facilities.

RMX 86 (Intel Manual f) is Intels' real time multi-tasking executive for the 8086 microprocessor. As such it combined the facilities of a multi user operating system with a real time task scheduler. It had several advantages over the two other operating systems in that information was readily available on its capabilities and it was also already in use in another project in the department, with favourable results. It also provided compilers for Pascal, Fortran and PL/M-86, as well as an assembler and various utilities. Its drawbacks were its very large size, it requires about 140K byte of ram, and it needs a hard disk to function efficiently. Despite these drawbacks it was decided to use RMX 86 as the operating system since it was felt that the advantages outweighed the drawbacks.

3.1.2 RMX 86 Operating System Features.

This section is included to familiarise readers with some of the more important features of the RMX 86 operating system as it was implemented in the HVDC development system. More information can be found in reference "Introduction to the RMX86 Operating System".

As stated above RMX requires a hard disk to operate efficiently, it is however capable of operating with a pair of floppy disks, although this slows the speed of the compilers and other programs using the disk because the data transfer rates from a floppy disk are much slower than those that can be achieved from a hard disk.

As well as supplying compilers for Pascal, Fortran and PL/M-86 a number of utilities, such as a file editor and object file linkers and locaters are also available. This permits the source code routines to be split into a number of small modules, each of which can be written and debugged separately. The individual modules are then combined by the linker and references to variables resolved between modules, while the locator sets the final run-time addresses.

More importantly, a library of simple interface routines is provided that permits a users program to be linked in to the operating system. These take the form of a number of procedures that may be called from a user program to access system resources. They permit a user program to, amongst other things, transfer data to and from disk files at any time. This has proved to be extremely useful since it allows data from the convertor captured by the microprocessor to be stored for permanent reference or subsequent analysis. There is, of course, a trade-off. Use of the libraries requires that extra code be linked to the user program, resulting in more memory space being required to run it. Compared with the advantages offered this is only a very minor problem.

Finally the most important feature of RMX is that it is user configurable. That is the component parts are supplied with a number of user selectable options. The user then builds an operating system by selecting the features he requires. This permits the customising of RMX to fit any target system. RMX does make some assumptions about the hardware available, only Intel made support components are accepted, and must be used. This is not a problem although the processor that runs the operating system is a Matrox MBC 86-12-128, as this has all the required support components.

3.1.3 Overheads Imposed by the Operating System.

RMX 86 is intended as a real time multi-tasking executive program and as such is capable of scheduling tasks to run in response to external real-time stimuli. In practice, the time taken to switch tasks through the operating system is too long and this feature is not used by the HVDC system programs. The size of the operating system also provides another major overhead.

In the interests of execution speed it is desirable that each processor in the HVDC system only execute programs in its own on-board memory. If a processor is forced to go to off-board memory for instructions then the bus usage climbs dramatically, and this in turn slows the other processors in the system. For this reason memory on the board running the operating system must be reserved for the HVDC interactive

programs. This in turn means that the whole of RMX cannot fit in the remaining memory space, and the overflow is contained on an external 256k byte ram card. The remaining space on this card is not used by the HVDC system, but is reserved for the use of the RMX 86 compilers and utilities.

By partitioning the memory space in this way into separate areas for the operating system and the HVDC system it is possible to switch between running one or the other without corrupting the code of either. This speeds development and debugging of the HVDC system, but requires more memory than would otherwise be needed.

3.2 Languages.

The choice of language is constrained by the operating system chosen, but the final choice made is still of great importance as it will directly affect the execution speed of the HVDC programs.

3.2.1 Choice of Languages.

Programs written in assembly language generally execute fastest, but the mass of detail that the programmer must handle slows the production of good code. For this reason the higher level languages that are more commonly used on larger mini and main frame computers are also often used on microprocessors. Unfortunately most high level languages such as Pascal of Fortran are not well suited to microprocessor use. They tend to produce much larger amounts of executable code for a given algorithm than an assembly language would, and the programmer often loses the ability to directly access specific memory or i/o resources. There are several newer languages which retain the advantages of the high level languages while overcoming some of the disadvantages. Examples of these include the 'C' language and PL/M-86 (Intel Manual k). The department already had compilers for PL/M-86 running on both the microprocessor development systems and the VAX 11/750 computer. RMX 86 also provided a compiler for PL/M-86 and for this reason PL/M-86 was the language chosen for use in the project. It has been used extensively in the writing of the interactive routines, and also in the initial development and testing of some of the real time routines.

3.2.2 Features of PL/M-86.

PL/M-86 is described by Intel as an application language. It provides the structured, modularised, procedure orientated advantages of a high level language while still allowing the programmer to communicate with specific memory and i/o resources when necessary. It supports basic arithmetic functions for unsigned eight and sixteen bit numbers, as well as for sixteen bit signed integers and thirty two bit floating point numbers.

The compilers for PL/M-86 provided by Intel also offered several advantages. The object modules produced can be combined together with those produced from assembly language sources and the compiler has built-in optimisation features that result in the compiled code being little longer than the code that would be produced from assembly language. Experience has shown that a PL/M-86 implementation of an algorithm produces only about 25% more code than the equivalent assembly language implementation, and has a correspondingly longer execution time. This overhead is a small price to pay for the much increased speed of code generation that the use of PL/M-86 offers.

3.3 Software Development Techniques.

Where ever possible a structured approach has been adopted for the development of the HVDC software. All interactive programs are written as small modules containing only one or two procedures. These modules are then combined by the linker to form the executable program. This permits easy expansion and testing of code as changes from one step to the next can usually be confined to a single module.

All code produced can be divided into two broad categories, that code that is executed in response to real time events, and that code that does not need to run real time. The first group comprises the interrupt service routines, together with the programs that process the data provided by the interrupts. The second group contains the interactive routines that permit the operator to monitor the operation of the convertor and control to a certain extent the operation of the first group.

In general the execution times of the first group are critical and all of these routines are written in assembly language to provide the shortest possible execution times. The execution times of the routines in the second group are unimportant, and since they comprise the bulk of the programs written, they have been written in PL/M-86 in order to speed program development.

4.0 Data Acquisition Sub-System.

All the sub-systems that are described in this thesis will only operate correctly if the data supplied to them is both accurate and up to date. Thus the operation of the data acquisition sub-system is of vital importance to the successful operation of the entire HVDC development system. This chapter starts by describing the data requirements of each of the major sub-systems and then proceeds to describe how these requirements have been met.

4.1 Sub-system Data Acquisition Requirements.

4.1.1 Control Sub-system.

The data required from the convertor for the control system consists primarily of the dc line current, the minimum delay and extinction angles recorded during the previous cycle and, when operating in constant extinction angle mode, the value of each valve's extinction angle recorded during the previous cycle.

Values of minimum delay and extinction angle can only be updated at each valve turn-on or turn-off whereas the dc current is a continuously varying quantity, and the most up to date value should be used. In practice there is always a small delay between the instant when the value was measured, and when it is used in the control calculation, and this may have a detrimental effect on controller accuracy. The use of separate control and data acquisition processors minimises this effect however, as the value may be updated by the data acquisition processor during the course of the control calculations.

The controller also requires information from the operator with regard to the desired convertor operating point, and provision is made through the interactive processor to permit the operator to change all possible control parameters. Provision has also been made to permit the operator to enable or disable selected parts of the control algorithm during execution. All of the data acquisition tasks required by the control system are run on the data acquisition processor. This allows the control processor to respond to the control interrupts as and when required.

4.1.2 Fault Simulation and Detection Sub-systems.

Unlike the control sub-system the fault simulation sub-system requires relatively little information from the convertor. The data required being limited to the commutating voltage zero crossing instants, which are required to synchronise the operation of the fault simulator with the convertor. As in the case of the control sub-system, detailed data is required from the operator in order to specify the fault condition to be simulated and, again, provision has been made for this in the interactive software. As will be described in chapter 6 the fault simulation tasks are actually run on the control processor. Since the fault simulator must respond instantly to each zero crossing interrupt, the fault simulator's data acquisition task is also run on the control processor.

The fault detection sub-system, which is totally independent of the fault simulation sub-system, requires the convertor valve turn-on and turn-off sequences and the controller firing pulse sequences. The data acquisition task to provide the on/off sequences is run on the fault detection processor. This has been done to ensure that the information is available to the fault detection task as quickly as possible.

4.1.3 Interactive Analysis Sub-system.

The Interactive analysis package operates independently of the fault simulation and control sub-systems in that no information from them is passed across to the analysis routine, but the simulation and control routines do trigger an enhanced data capture process to provide sufficient information for a detailed analysis. Since the enhanced data acquisition task is based on the standard task it is also run on the data acquisition processor. This enhanced data acquisition process will generally be refered to as the storage mode of operation.

4.2 Conventional and Storage Modes.

The data acquisition process being described is capable of producing extremely large quantities of data. In an attempt to reduce this data to a manageable size, during normal operation only the data captured during the most recent cycle is stored. During fault simulation or whenever a major controller parameter is changed the amount of data needed by the analysis package is substantially increased. Instead of over writing the data values as the new values become available and discarding the data buffer entries as they are processed, all information is transfered into a special data base.

This operation is triggered by the fault simulation or control sub-systems several cycles, the exact number being set by the operator, before the event takes place. Detailed recording continues for about twelve cycles after being triggered. In this way a complete record of the convertor's operation from before the event occurred until recovery is complete is obtained.

This enhanced operation has the disadvantage that a large amount of data is produced, typically in excess of 50K bytes, and this is why recording in this mode is limited to twelve cycles.

4.3 Convertor Angle Measurement.

From the data acquisition requirements listed above it is apparent that the measurement of delay, commutation and extinction angles are of major importance.

4.3.1 Detection of On/off and Zero Crossing Instants.

Logic signals indicating the conduction state of each valve in the bridge are available from the convertor. These have been derived from the valve anode to cathode and gate to cathode voltages and provide an indication within 1.5 microseconds of the actual switching taking place (Arrillaga and Hisha, 1979). Logic signals representing the states of the three phase commutating voltage waveforms are provided from the zero crossing detectors located on the angle measurement board, the operation of which was discussed in the hardware chapter.

In each case the instant of transition is determined in a similar manner. The incoming signals are applied to one side of a digital comparator, either three or six bits wide, and are also made available to the microprocessor through a corresponding data port. On the other side of the comparator a set of latches hold the states measured after the last transition. Thus while the inputs remain unchanged the comparator has identical sets of inputs and indicates this equality on its output. When one or more input lines change the equality is removed and the change in comparator output generates an interrupt for the data acquisition processor. The processor immediately responds by reading the new input states and writing them back to the latch. This restores the equality and removes the interrupt request.

As well as the data acquisition processor, the fault detection and control processors are also interrupted by the on/off and crossing transitions. This permits the fault detection and fault simulation software to obtain the information required. The response to these interrupts will be covered in the appropriate chapters.

4.3.2 Data Acquisition Interrupt Service Routines.

Fig 4.1 shows the structure of the service routine.

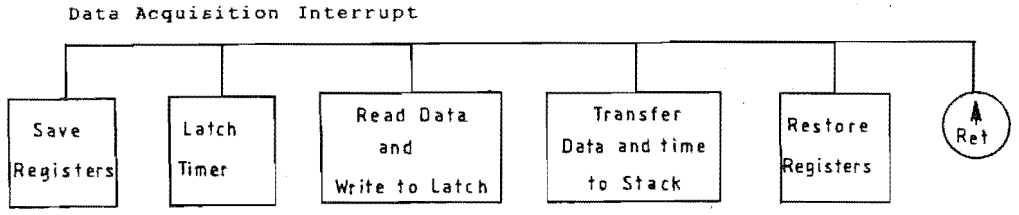


Figure 4.1 Data Acquisition Interrupt Routine

The first step, standard procedure in all interrupt service routines, is to save the contents of any registers used by the service routine. The system clock is then latched, capturing the time at which the interrupt, and hence the valve turn-on or turn-off, occurred. There is of course a small time delay between the interrupt generation and the latching operation, but this time delay occurs at every interrupt and is of fixed duration. Since the method being described relies on the difference between successive interrupts, the delay does not affect the accuracy of the measurements.

The new valve conduction state is then read from the appropriate data port and immediately written back to the comparator on the angle measurement board to clear the interrupt request. The data is then placed onto the top of a circular buffer together with the time and a byte identifying the buffer entry as coming from a valve on/off event.

A virtually identical interrupt service routine is run in response to the crossing interrupts, except that the identification byte in each buffer entry is changed. The same buffer is used by both routines and this consequently contains a record of convertor activity in chronological order.

Processing in the interrupt service routine has been deliberately kept to a minimum in order to keep the interrupt service time as short as possible. This is necessary because once one interrupt occurs it is not possible to respond to a subsequent interrupt without losing data until the first has been processed. Thus if a zero crossing was quickly followed by a valve switching then the time recorded for the switching will be increased by the length of time taken by the crossing interrupt service routine. This can introduce a small error into the measurement process.

4.3.3 Angle Calculation.

The buffer entries are cleared by the data acquisition main line program which runs whenever an interrupt is not being serviced. Each entry in the data buffer consists of four bytes, the first is used to identify the data following as coming from either a crossing interrupt or a valve on/off interrupt. The next byte in the entry contains the new crossing or valve conduction state, depending on the source, and the final two bytes contain the time at which the interrupts occurred. The entries are decoded and a corresponding angle is calculated.

The angle being calculated is measured as the difference in time, as determined by the system clock, between a pair of transitions. For example, an event such as the commutation from valve V5 to valve V1 allows the delay angle for valve V1 to be measured as the time between the previous crossing C1 and the valve V1 turn-on. Similarly the commutation overlap angle for valve V5 is measured as the time between the turn-on of valve V1 and the turn-off of valve V5. The time of the valve V5 turn-off

will also be stored and used to determine the extinction angle for valve V5 when the next crossing C4 is detected. Thus at each turn-on a corresponding delay angle is calculated, at each turn-off a corresponding commutation angle is calculated and at each zero crossing a corresponding extinction angle is calculated. Fig 4.2 shows the structure diagram for the mainline routine that performs this task.

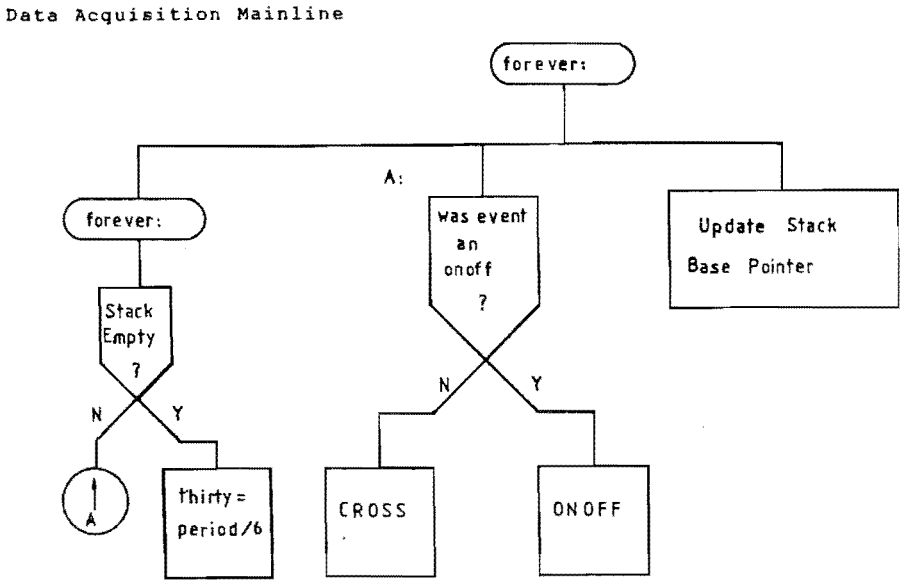


Figure 4.2 Data Acquisition Mainline

As each new entry on the data buffer is detected, the first byte is examined and either the crossing routine or the on/off routine is called to process the entry. Fig 4.3 shows the operation of the on/off routine.

In the on/off routine a valve turn-on is detected by performing a logical exclusive or of the new conduction pattern and the previous pattern. This result is the logically anded with the new pattern to identify any valve that turned on. This process produces a byte with a bit set high in a position corresponding to the valve. This is then decoded to identify the number of the valve that turned on. This valve number is then used as a pointer into an array of valve turn-on times, and the new value for this valve is transferred from the data buffer to the storage array, overwriting the previous value.

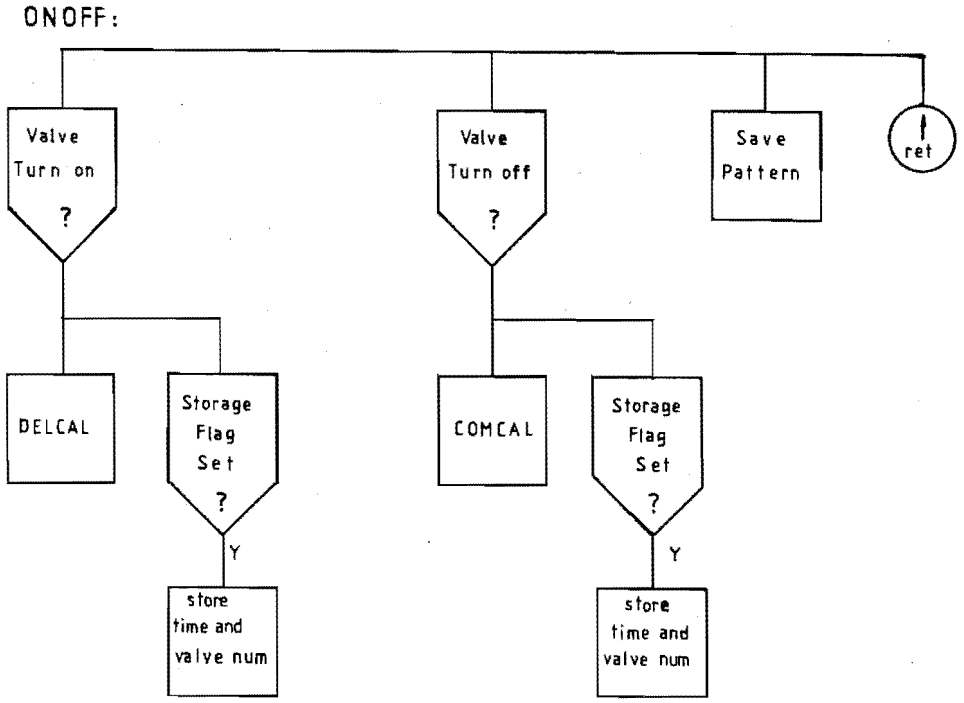
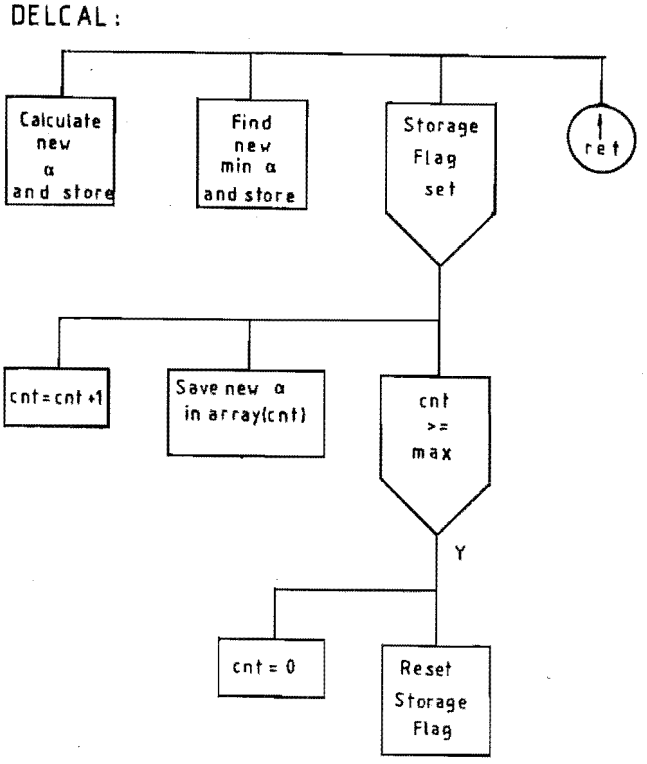


Figure 4.3 Data Acquisition Onoff Routine

Once a turn-on has been positively identified, a subroutine is then called to calculate the delay angle associated with the turn-on. This is subroutine Delcal and its operation is shown in fig 4.4 .

In Delcal the delay angle is calculated as $\alpha i = Ci - ONi$, the valve number i being passed across from the on/off routine. Once the new angle is determined it is saved into the delay angle array, again overwriting the previous value. This array is then searched for the new minimum value of angle which is stored for use by the controller. If the storage mode is enabled then the new delay angle just calculated is also saved into a separate database for subsequent use in the analysis routines. Due to limitations on available storage space, only twelve cycles of data are recorded for subsequent analysis. A counter (cnt) is maintained, and when the storage array is full, the storage mode flag is reset, disabling subsequent detailed data storage.



Data Acquisition Delcal Routine

Figure 4.4

On returning to the on/off routine at the completion of Delcal, the storage mode flag is re-examined and if it is set the time that the turn-on occurred and the number of the valve in which the turn-on occurred are also saved for subsequent analysis.

After processing the valve turn-on, or if no turn-on was detected, the on/off routine tests for any valve turn-offs. A similar technique to that used for the turn-on is used except that the exclusive ored product is anded with the old pattern rather that the new one. The valve involved is identified in the same way and the time is transferred from the data buffer to the turn-off time array overwriting the previous value. The corresponding commutation angle is the calculated in subroutine Comcal, shown in fig 4.5 .

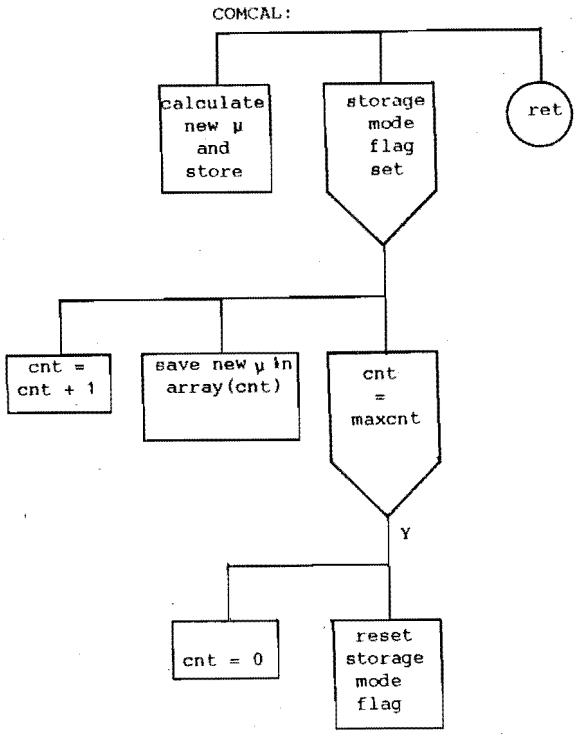


Figure 4.5 Data Acquisition
Comcal Routine

In this routine the commutation angle μi is calculated as μi = ONi+2 - OFFi. The new value is used to update the commutation angle array, although no minimum value is calculated as it is not needed by any of the sub-systems. Finally in the Comcal routine the storage mode flag is examined and if set, the newly calculated value of commutation angle is saved for analysis.

Again, on return to the on/off routine the storage mode flag is re-examined and if it is set then the turn-off time and the valve number are also stored for analysis. Finally the on/off routine stores the new conduction pattern for use next time before returning to the main line. The mainline then completes by updating the buffer pointers before waiting for the next entry to occur.

The procedure for a voltage crossing is similar and is shown in figs 4.6 and 4.7. Routine Cross is called by the main line to process the entry and this is presented in fig 4.6 .

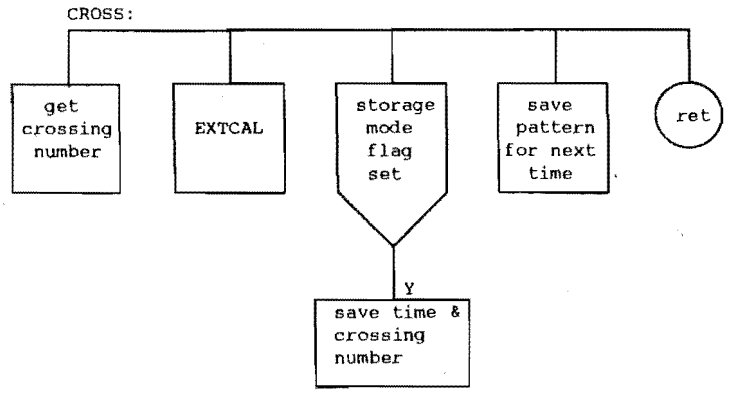
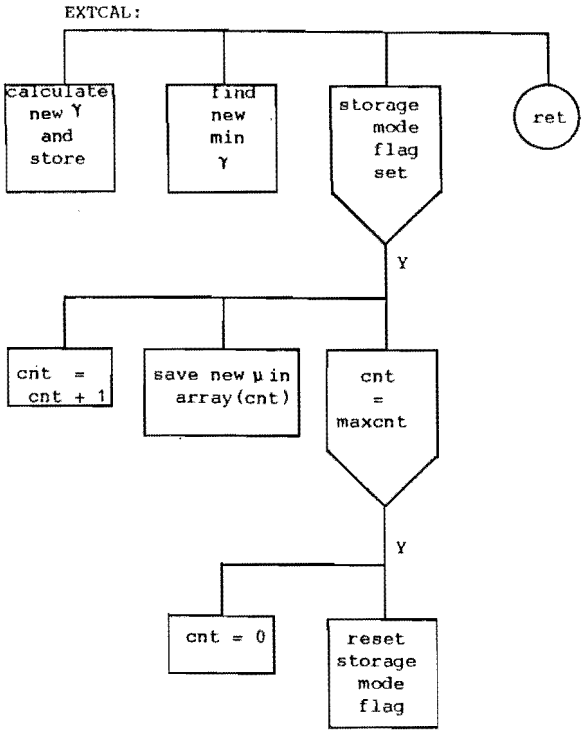


Figure 4.6 Data Acquisition Cross Routine

The crossing number is identified in much the same manner as were the valve numbers, and the time of the crossing is transferred from the data buffer to the crossing time array. The routine Extcal is then called to calculate the corresponding extinction angle and this is shown in fig 4.7 .

The new extinction angle γ_i is calculated as $\gamma_i = OFF_i - C_{i+1}$ and the new value is placed in the extinction angle array. The new minimum extinction angle is then calculated and saved for use by the controller. Again, if the storage mode is enabled, then the new value of extinction angle is also saved for use by the analysis routines and on return from Extcal the Cross routine also saves the crossing time and number if the storage mode is enabled. Cross also completes by saving the new crossing pattern for use next time it is called.



Data Acquisition Extcal Routine

Figure 4.7

4.4 Direct Current Measurements.

As described in section 4.1.2 the dc current is required by the control software and can also be used by the monitoring system. The signals supplied from the convertor are analogue voltages and so some form of analogue to digital conversion is needed before they can be processed by the microprocessor.

The sample rate of the a/d convertors is controlled by a timer. This timer is set by the operator to interrupt at specified intervals. The response to the interrupt, as shown in fig 4.8 is to restart the timer and start the conversion by first latching the sample and hold circuits and then starting the actual a/d convertors.

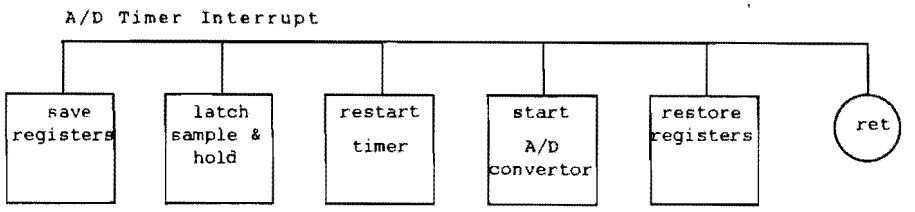


Figure 4.8 A/D Timer Interrupt Routine

A second interrupt occurs when conversion is complete and the response to this is shown in fig 4.9. The interrupt request flip flop is cleared and the values read in and stored in memory. Two options are provided, the value stored in memory can be either the value read directly from the a/d convertor or can be the average of the last sixteen measurements. The option to be used is selected by the operator and may be changed during operation.

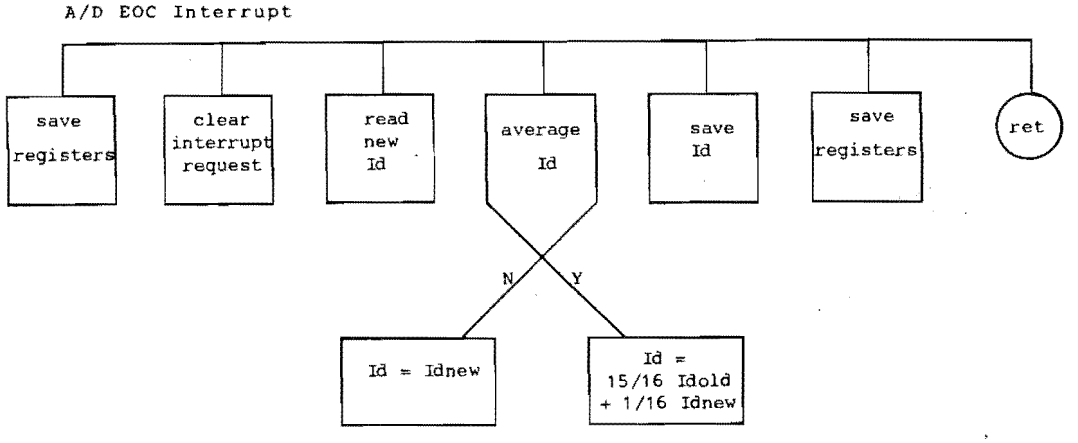


Figure 4.9 A/D EOC Interrupt

4.5 Resolution and Accuracy of Measurements.

4.5.1 Angle Measurements

One source of error in the measurement process has already been discussed in section 4.3.2 . The effect of this error is to impose a limit on the minimum angle that can be measured. In the system under discussion this minimum angle is approximately 1.2 degrees. Commutation overlaps of

shorter duration appear as zero degrees since the turn-on and turn-off appear to have occurred simultaneously. The delay and extinction angles are only affected over a small range of values about the 0, 60, 120, and 180 degree points, and as operation near these points is not usual under normal operating conditions the problem is not serious. Even when it does appear the effect is merely to increase or decrease the angle by no more than 1.2 degrees.

The resolution of the measurement is determined by the system clock. Since a sixteen bit timer was the largest convenient size available, the maximum frequency that can be used is limited to 6.55 MHz. If a higher frequency is used then the timer will overflow and wrap around before the end of the cycle. If this occurs a large angle will be measured incorrectly as having a much smaller value.

In this system a frequency of 1.8MHz is used, which has the advantage that each 'tick' is of 0.01 degrees duration for a 50 Hz system, and the values obtained by the angle measurement routines may be conveniently displayed without further conversion. The values of the angles are only displayed to one decimal place and so any fluctuation of the display is due to fluctuations in the convertor and is not a result of limited resolution.

4.5.2 DC Current.

The major requirement in the measurement of the dc current is for good accuracy and adequate dynamic range. Ideally the a/d convertor should have a resolution better than the rest of the measuring circuit. For example an eight bit convertor typically has an accuracy of one least significant bit. If its full range is to be used this results in a resolution of less than ±0.5%, but the convertor is incapable of measuring any over-current or over-voltage during fault conditions. If the range 0 to 100 was used then over-currents up to 2.55 p.u. can be measured, but the resolution drops to ±1%. For this reason it was considered that an eight bit convertor would be unable to give adequate over-range capability and still maintain acceptable resolution. A ten bit convertor would yield adequate results, but using a twelve bit convertor allows the range 0 to

1000 to represent the current 0 to 1 p.u., and provides an over range capacity of 4 times. This allows any value to be displayed directly on the operators screen without requiring any conversion, and provides a nominal measurement resolution of ±0.1%.

5.0 Control Sub-system.

The control sub-system is only one of four major sub-systems incorporated in the HVDC development system. It is however of major importance in that it must maintain the stable operation of the convertor before some of the other sub-systems can operate. It must be restated at this point that the aim of the project was not to implement a 'state of the art' control system on a microprocessor, but rather to investigate as many applications of microprocessors to HVDC systems as possible.

In order to familiarise readers with the basic principles of HVDC control, this chapter starts by reviewing existing analogue techniques before quickly describing earlier attempts at direct digital control. It then describes the implementation and operation of the proposed microprocessor based system and how the basic control task may be expanded to incorporate other useful functions.

5.1 HVDC Control

5.1.1 Basic Control Principles.

Since the purpose of an HVDC link is to supply controlled power from one ac system to another, then the obvious task for the controller is to control the dc power being transferred. This may be done by controlling directly either the dc voltage or current.

The dc voltage is rarely directly controlled however (Kimbark, 1971), (Jotten et al., 1978) since a constant voltage characteristic would produce a considerably higher current in the event of a dc line fault than would result if a constant current controller was used. Thus it is normal practise to use a controller with a constant current characteristic.

Several other considerations affect the overall control characteristic. The first of these is that the current cannot be increased indefinitely, because eventually the maximum dc voltage available for the given system will be reached. This occurs when the convertor is operated with zero delay angle and this imposes the $\alpha = 0$ line on the convertor characteristic shown in fig 5.1 . This first limit is a naturally enforced, but the second limit $\gamma = \gamma min$ must be enforced by controller operation.

This extinction angle limit must be provided to ensure that commutation completes, and the outgoing valve recovers its blocking ability before the commutating voltage reverses polarity. If insufficient time is provided for this to happen then a commutation failure will occur, resulting in the incoming valve being turned off, leaving the outgoing valve in conduction.

The actual operating points of the two convertors are usually set with regard to the reactive power consumption of each unit. The reactive power consumed by the rectifier will be at a minimum when it is operated on or near the line $\alpha = 0$, although if the rectifier were to be operated on this line the control over the current would be lost. Since the invertor end is a power source for the receiving system, the invertor should ideally provide reactive power as well as real power. This is of course impossible as the convertor always consumes reactive power and so it is usually very important to reduce the reactive power consumption in the invertor to as small a level as possible. For this reason the invertor is always operated on the $\gamma = \gamma min$ line whenever possible. With the invertor operating on this line the rectifier end must assume current control leading to the link characteristic shown in fig 5.2 .

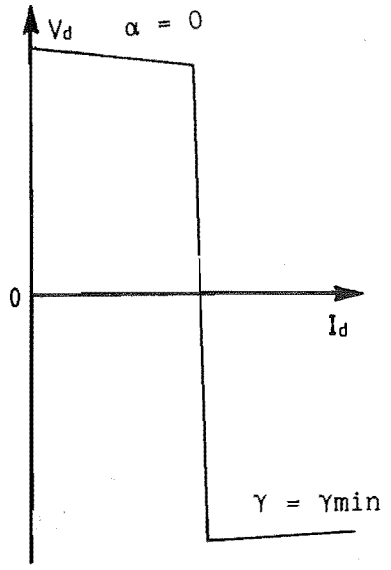


Figure 5.1 Convertor Load Characteristic

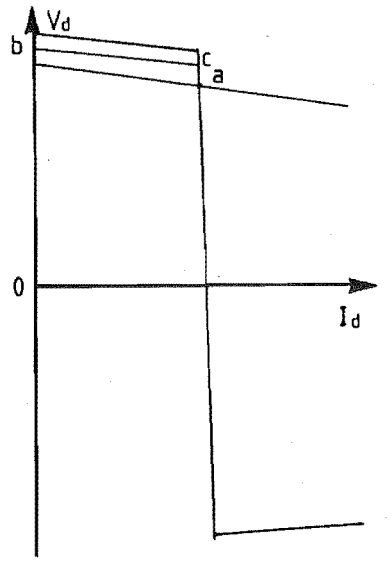


Figure 5.2 Basic Link Load Characteristic

This is obviously only useful as long as the invertor voltage is below that of the rectifier. Since the operating point is deliberately chosen to be as close to the knee in the characteristic as possible in order to minimise the rectifier reactive power consumption then any increase in invertor voltage or decrease in rectifier voltage results in a loss of controllability and the load current decreases to below that needed to hold the valves on.

This problem is easily solved by providing a constant current controller as well as the constant extinction angle controller in the invertor. This gives the characteristic shown in fig 5.3 . The current order for the invertor is deliberately set some amount, the current margin ΔId, lower than the rectifier. As a result the invertor sees too high a value of dc current and progressively increases its delay angle in an attempt to reduce the dc current. The delay angle is increased until the extinction angle reaches its minimum permitted value and the constant extinction angle control takes over and maintains the invertor on the $\gamma = \gamma min$ line, resulting in the HVDC link operating at point A on fig 5.3 .

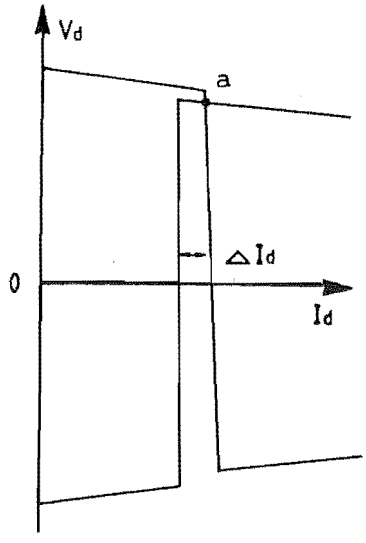


Figure 5.3 Full Link Load Characteristic

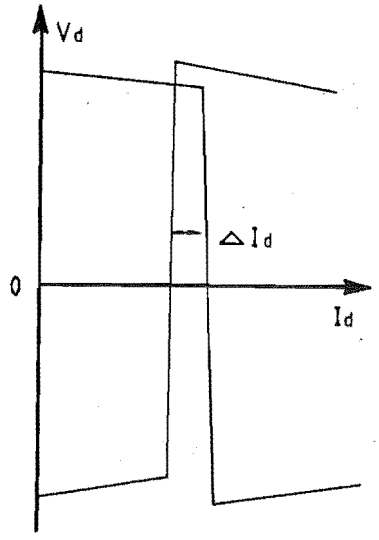


Figure 5.4 Invertor Current Control Load Characteristic

During an increase in invertor voltage or a drop in rectifier voltage the constant current loop of the invertor will take over, as shown in fig 5.4 . In this case the reactive power consumption of the invertor is no longer a minimum, but this is of less importance than maintaining the operation of the link.

In addition, a minimum delay angle loop is often provided for the rectifier to ensure that the valves are sufficiently forward biased to take over conduction as soon as they are fired. This produces the modified characteristic b-c in fig 5.2 The basic equivalent circuit of a dc transmission system is shown in fig 5.5.

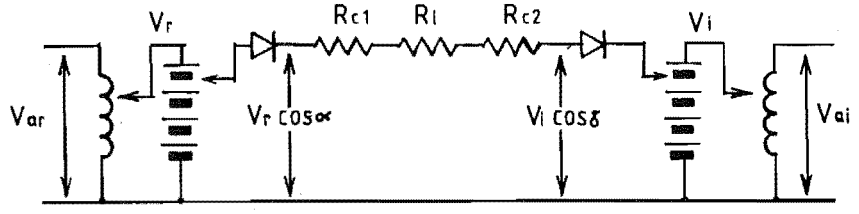


Figure 5.5 DC Link Equivalent Circuit

The rectifier voltage is given as Vr COSα , while the invertor terminal voltage is given as Vi COSϒ . The resistances Rc1 and Rc2 represent the voltage lost in the commutation overlaps at the rectifier and invertor ends respectively, and Rl the line resistance. From this it can be seen that the dc current will be determined by the difference in voltage between the two ends divided by the total link resistance, ie.

$$Id = \frac{(Vr\ COS\alpha\ - Vi\ COS\gamma\ )}{Rc1 + Rl - Rc2}$$

To control the dc current therefore, it is necessary to control either the convertor terminal voltages or the link resistances. As the link resistances are fixed, it is necessary to control the rectifier delay angle or the invertor extinction angle, and since, as described above, the invertor extinction angle is usually fixed, it is normal to control the dc current by controlling the rectifier delay angle.

5.2 Analogue Implementations.

5.2.1 Individual Phase Control.

The earliest technique used to implement the control requirements outlined above was the individual phase control, in which the individual delay angles were calculated using the commutating voltages and valve currents.

Implementation of the constant current control is relatively simple. The measured dc current is subtracted from the required current order and the error term used to control a variable phase shift circuit that varies the delay angle to the valves. This technique, in its most simple form, is shown in fig 5.6.

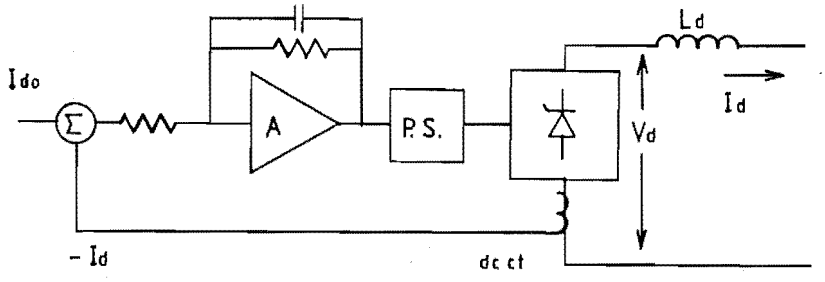


Figure 5.6 Constant Current Individual Phase Control

The implementation of constant extinction angle is somewhat more complicated. It is normal to set the minimum extinction angle γ_n to provide sufficient time for the outgoing valve to recover its blocking ability allowing for any fluctuations that might occur in either the dc current or the commutating voltage. It can be shown that the ignition angle required to meet this is given by

$$-\sqrt{3}E_m \cos\omega t = \sqrt{3}E_m \cos\gamma_n - 2X_c I_d. \quad \text{(Kimbark, p169)}$$

This equation must be solved individually for each valve in the bridge, and a simple analogue computer is commonly used. A simple representation of this is shown in fig 5.7.

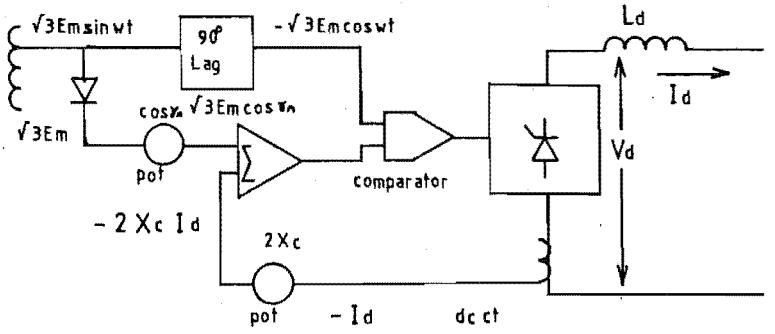


Figure 5.7 Constant Extinction Angle Individual Phase Control

In practice these two circuits would be considerably modified to include many subsidiary loops, but the basic forms shown here are representative of the predictive techniques used commercially prior to 1970.

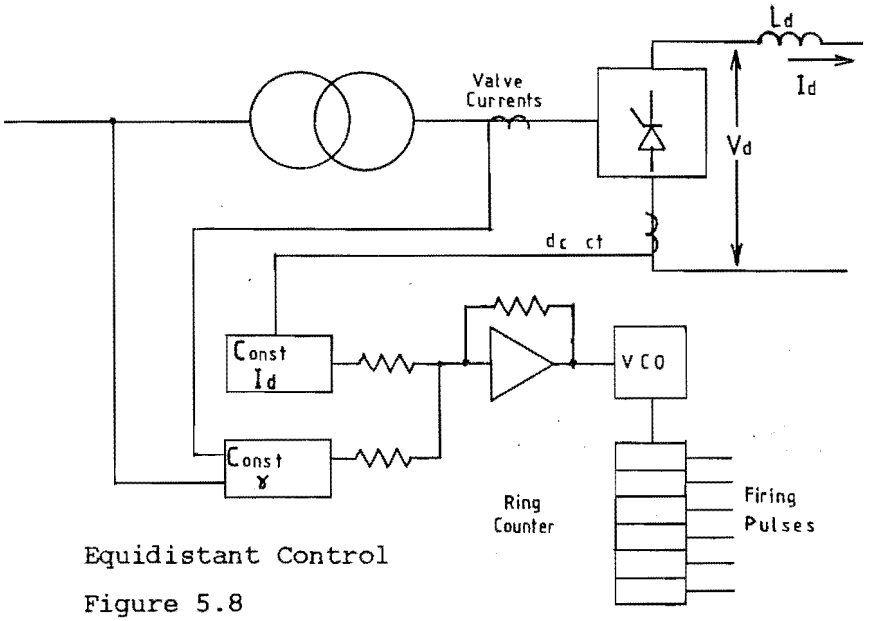
Alternative schemes have been proposed (Hingorani and Chadwick, 1968), (Machida and Yoshida, 1967) based on the closed loop control of extinction angle. In these schemes, the individual extinction angles are measured and the difference between the measured and ordered value is fed back to control the next delay angle.

All of these individual phase control schemes suffer from a particular problem in that even during normal operation and especially during disturbances, non characteristic harmonics produced by the convertor distort the commutating voltages. This distortion in turn affects the controller, causing it to incorrectly place the subsequent firing pulses which in turn may increase the levels of non characteristic harmonics produced. If the ac system has a resonance near one of the non characteristic harmonic frequencies then instabilities can result (Ainsworth, 1968b).

5.2.2 Equidistant Control.

This form of control was originally proposed to overcome the problems associated with the individual phase controls described above (Ainsworth, 1968a). The basic principle is that the firing pulses should be spaced at exact sixty degree intervals, thus removing the dependence on the distorted commutating voltage waveforms. The basic format of this type of control is shown in fig 5.8.

The error between the measured dc current and the ordered value is used to control the frequency of an oscillator nominally running at six time ac system frequency. The output of the oscillator is then used to clock a shift register that delivers six equally spaced firing pulses every cycle. If the two currents are equal then the oscillator frequency will be exactly six times that of the ac system. However if the measured value should change for any reason then the error term changes the oscillator frequency, resulting in an advance or delay in the firing pulses which in turn restores the current to the desired value. As shown in fig 5.8 extra loops, such as minimum extinction angle may be added to the basic system to produce the desired controller characteristic.



Equidistant Control

Figure 5.8

Variations and enhancements of this basic form have been described by various authors (Ekstrom and Liss, 1970), and this method or one of the variants are now used in all new HVDC schemes (Jotten et al., 1978) (Uhlmann, 1975).

5.3 Direct Digital Control.

The fact that the HVDC convertor is suitable for direct digital control has long been recognised (Arrillaga and Galanos 1970b), (Reeve and Sevenco 1972). The requirement that the firing pulses be delivered at precise but discrete instants and that most of the important signals from the convertor can be readily represented in digital form has lead to many proposed schemes (Arrillaga, Galanos and Powner, 1970),(Reeve, Sevenco and Vali, 1972), (Arrillaga and Erinmez, 1971), (Arrillaga and Baldwin, 1974).

5.3.1 Logic Gate Based Schemes.

These were the earliest form of direct digital control system proposed. They provided a digital implementation of the equidistant algorithm described above. In general they consisted of a programmable counter driven from a clock that was phase locked to the ac system frequency. The timer was set to run for a nominal sixty degree interval under steady state conditions, and the starting count would be modified in some manner to provide the necessary delay angle phase advance or retardation. The exact methods of modifying the start count vary , but in general the constant current characteristic was implemented by using analogue circuitry to produce an error term which was then converted to some form of digital representation and used to increment or decrement the starting count appropriately. Constant extinction angle control was implemented by measuring the extinction angle error directly with a digital counter and using this to alter the timers' starting count.

The major drawback of these systems was the very large number of discrete gates needed to implement them , and the correspondingly complicated interconnection schemes required. This lead to doubts about the resulting reliability and was one of the major reasons for the lack of commercial interest shown in such schemes.

5.3.2 Mini/Microprocessor Based Schemes.

The development of the mini computer in the late sixties and the microprocessor in the mid seventies offered an alternative implementation technique, and schemes based on either small minicomputers (Shore and Freris, 1978) or microprocessors (Reeve and Giesbrecht 1978), (Arrillaga, Dewe, et.al., 1979) have been described in the literature. In general these schemes have used the microprocessor to replace some of the mass of logic circuits required by the earlier schemes. The control routines implemented have been generally of a fairly simple type and have made no attempt to exploit any other possible applications that the microprocessor might have. Indeed one of the advantages of using a microprocessor is that it should be possible to reduce the amount of hardware required by increasing the amount of software. However most existing schemes still seem to require large amounts of dedicated hardware in addition to the microprocessor (Dewan and Dunford, 1983), (Tso and Ho, 1981).

5.4 Microprocessor Based Control System.

5.4.1 Aims.

In the terms of this project, the actual performance of the control system is of less importance than a demonstration of the opportunities that a microprocessor based system offers. To this end, an extremely simple control algorithm has been implemented. This contains the major features of most modern HVDC control systems, but omits many subsidiary control loops normally added to optimise a particular convertor's performance in a particular situation.

The major aims of the control sub-system implemented were to demonstrate that the microprocessor is capable of maintaining control over the convertor and to investigate any possible advantages that a microprocessor implementation might offer.

5.4.2 Basic Control Algorithm.

The control algorithm in its simplest form consists of a constant current loop and both invertor and rectifier safety loops. The constant current loop is the major control loop while the two safety loops serve to protect the convertor at the extremes of minimum extinction and minimum delay angles. The actual constant current loop has a dual slope characteristic with the feedback gain being reduced whenever the delay angle falls below twenty five degrees or the extinction angle falls below forty degrees. This reduction in gain is necessary to ease the transition from the constant current loop to the constant delay or extinction angle modes of the convertor safety loops. If this is not done then an undesirably large limit cycle oscillation will develop at the transition point between the two loops. The reduced gain contains this limit cycle to within acceptable levels.

The rectifier safety loop takes over from the constant current loop whenever the minimum delay angle becomes less than the rectifier safety order. It operates to keep the delay angle above the minimum permitted value. The invertor safety loop acts in a similar manner on the extinction angle when the convertor is operated as an invertor. Fig 5.9 shows the structure of the main control algorithm implemented. Note that in fig 5.9 each loop has its own feedback gain factor. This factor is setup to a default value when the code is loaded, but the operator may change the values at any time.

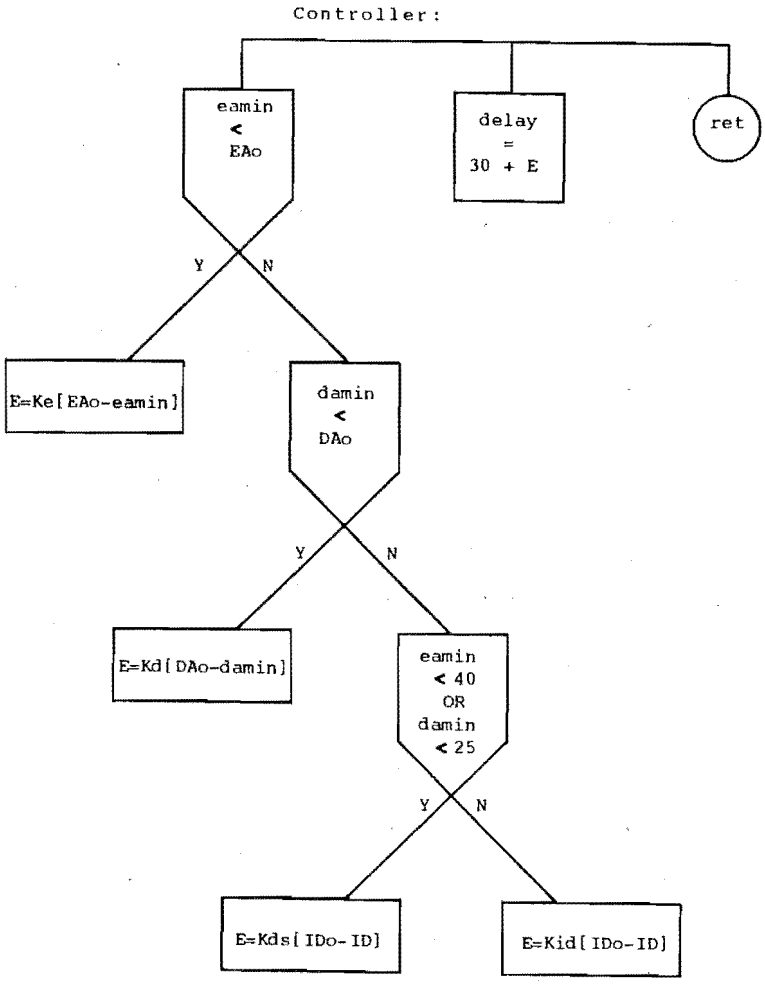
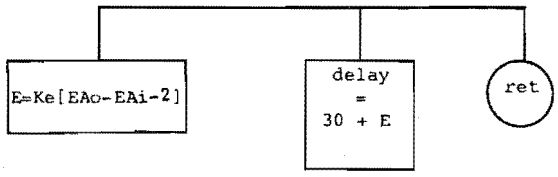


Figure 5.9 Control Algorithm

The serial nature of the microprocessor's operation is useful in the control calculation since it simplifies the switching between loops. Once a priority for each loop is decided upon it is simple to order the testing to implement this priority. Thus in fig 5.9 it can be seen that the minimum extinction angle loop has highest priority, with the minimum delay angle loop next highest. Following this is the dual slope section of the constant current loop and finally the constant current loop itself. In addition because effectively only one loop is executed during the calculation the time required for the calculation is relatively short. The actual calculation time varies from between 130 to 150 microseconds, with 60 microseconds of this being due to the multiplication and division step required by the adjustable gain feature.

In addition to the main control algorithm, a second control algorithm was also included. This takes the form of a constant extinction angle control and the operation of this loop is shown in fig 5.10.



Constant Extinction Angle Algorithm

Figure 5.10

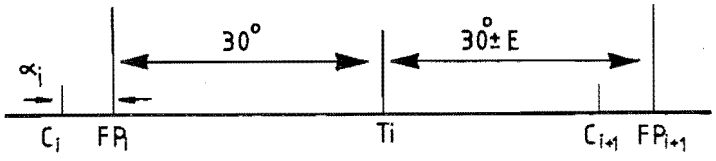
In this loop the value of the extinction angle recorded for each valve in the previous cycle is used to determine the correction term needed in the current cycle. Since it is the delay angle that is being directly controlled, the correction term for firing valve Vi is determined by the error between the extinction angle order and the measured extinction angle of valve Vi-2. This is because the point at which, for example, valve V3 is turned on determines the extinction angle of valve V1.

The controller is intended to have an equidistant characteristic, under which the firing pulses are spaced at sixty degree intervals during steady state operation and, as shown in figs 5.9 and 5.10, the control algorithm calculates a value 30±E, where E is the error term. This when combined with the rest of the control system timing ensures that when E = 0 then the equidistant requirement is maintained. The constant extinction angle control mode will also exhibit an equidistant characteristic, but only when the commutating voltages are perfect sinusoids of equal magnitude. This never happens in practise, and consequently the equidistant characteristic is lost.

It is worth noting at this point that the control algorithm described is not specific to microprocessors and could equally well be implemented in conventional analogue circuitry.

5.4.3 Microprocessor Implementation.

The control sub-system is run on one of the four iSBC86/12a single board computers that make up the HVDC development system. This implementation utilises the available hardware resources on the board. The basic timing diagram for the control sub-system's execution is shown in fig 5.11.



Microprocessor Control Timing Diagram

Figure 5.11

At point FPi, a time αi after crossing Ci, the control processor receives an interrupt and it responds by first firing valve Vi and then starting a timer to run for thirty degrees. During this thirty degree interval the processor is free to calculate the firing delay using the algorithm shown in fig 5.9. When the timer reaches the terminal count at point Ti it reinterrupts the processor, which responds by restarting the timer with the value of delay just calculated. As stated in section 5.4.2 the delay has a nominal value of thirty degrees. Thus when the timer again reaches terminal count, at point FPi+1, and interrupts the processor a total of sixty degrees have elapsed, and the next valve Vi+1 is fired and the cycle repeated. Thus under steady state conditions the equidistant firing requirement is achieved.

Should the delay angle need to be increased or decreased to maintain the constant current requirement then the calculated variable delay will change from the nominal thirty degree value, resulting in an advance or delay in firing for the next valve, but the delay settles back to the nominal value as the dc current reaches the desired value.

Using the above technique it is , in theory, possible to advance or retard the delay angle by up to thirty degrees per firing, or up to 180 degrees per cycle. This is considerably faster than any connected ac system could withstand, and indeed the inductance in the dc line will limit the possible rate of change to considerably less than this, but it allows the controller to take rapid corrective action if a disturbance occurs.

The method has the advantage that it is simple, and by using the external timer, the cpu is free to run other important tasks. Since the iSBC86/12a processor board has three timers available, no extra hardware is required. It does have the disadvantage however that the control calculation is performed between thirty and sixty degrees before the valve is fired, and any change in the convertor during that time will not be reflected in the positioning of the firing pulse. In practice the results obtained indicate that this is not a major problem, since the controller will react to any changes within 1/3 of a cycle.

5.4.4 Interactive Capabilities.

The operation of the controller described in the previous sections can be made fully automatic, requiring no input from an operator and relying on predefined values in the microprocessor data base to set the convertor's operating point. However, in this form the microprocessor implementation offers no real advantage over an analogue implementation, and in some areas could even be considered inferior. By introducing interactive communication between the control sub-system and an operator it is possible to exploit the flexibility that the microprocessor implementation offers.

5.4.4.1 Operator Controllable Features.

The first, and most obvious, parameters to be made operator controllable are those variables that directly affect the convertors operating point. In this system these are the dc current order and the minimum delay and extinction angle orders, and these may be changed from the operators terminal at any time.

The second group of parameters to be made operator controllable are the various control loop feedback gains. The software sets up nominal values for these at system startup, but these may be subsequently changed by the operator if required. This is particularly useful during development when the effects of changing feedback gains on the convertors operation can be readily assessed. Finally in the controller a series of 'option control' bytes are provided to allow the operator to select various options while the convertor is operating.

Switching between the constant current and constant extinction angle loops is controlled in this manner. This permits on-line on-load changes in the control algorithm to be made and opens up the possibility of adaptive control systems being introduced. These would adjust the type of control algorithm to suit the current states of both connected ac systems and should permit greater overall system stability.

As well as switching between major control types, individual loops inside a particular algorithm may be selectively enabled or disabled by the operator. This can be extremely useful as a development tool as it permits the optimisation of individual loops in isolation of each other, and it also allows the operator to study the interaction between different loops. In the system being described, the rectifier and invertor safety loops together with the dual slope section of the constant current loop may be enabled or disabled in this manner. In addition it is possible to select options that affect the data acquisition performed for the control sub-system.

One option provided allows the operator to select between the use of the instantaneous or an averaged value of dc line current in the control calculation. A second option allows the value of the thirty degree interval used in the control calculation to be selected from either the value measured from the ac system, or a nominal value that assumes a constant 50Hz system. Selection of the measured value permits the control system to track any slow frequency swings without developing a corresponding current error. Use of the measured value however makes the controller less stable during ac disturbances near the convertor terminals, as the distorted commutating voltages disrupt the measurement process. The final option that may be selected in this group is whether the change in current order should be made in a single step or as a gradual ramp. The ramped change is commonly used as the rate of change in current order can

be limited to the speed at which the governors on the generators can respond. The step change can, however, be very useful in assessing controller performance.

5.4.4.2 Convertor Information Display.

The various interactive options described above require that a comprehensive display of convertor and controller data be made available to the operator. Fig 5.12 shows a photograph of the standard control mode display on the operators screen. This display is updated approximately once a second, or at the operators request.

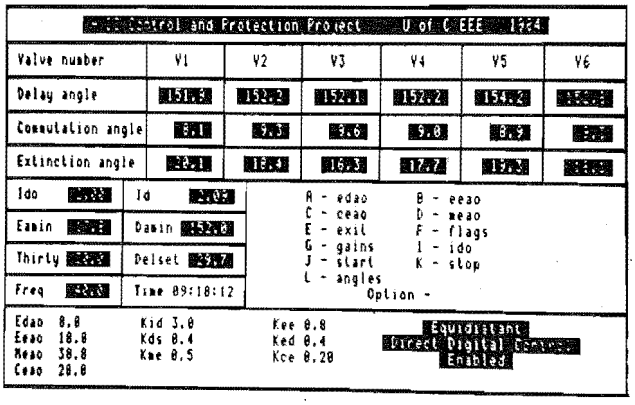


Figure 5.12 Operator's Screen Display

Information provided for the operator includes delay, commutation, and extinction angles for each valve in the bridge, measured and ordered values of dc current minimum delay and extinction angles, and the current values of the various feedback loop gains. A list of valid commands is also displayed for the operators convenience.

The standard display is not the only form available. By invoking the storage mode in the data acquisition system, described in section 4.2, it is possible to analyse the control sub-system in considerable detail. Fig 5.13 shows a typical result obtained.

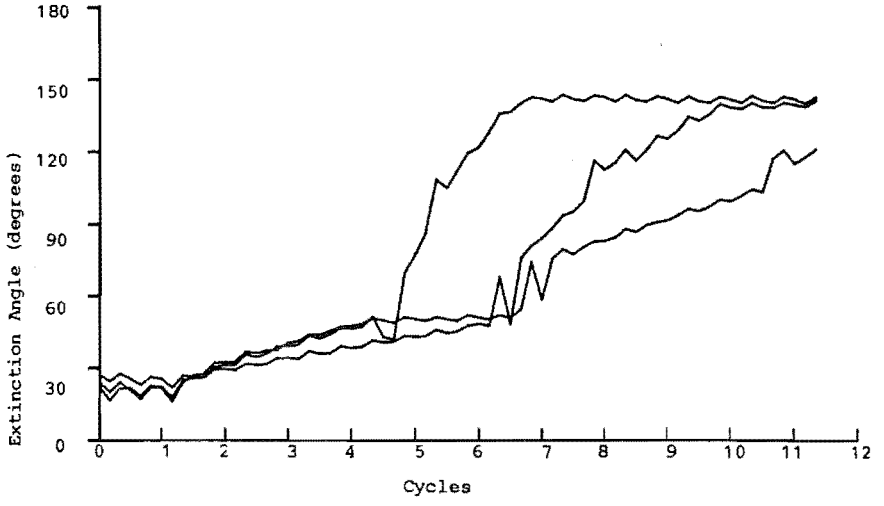


Figure 5.13 Controller Step Response

By analysing the data stored, it is possible to plot delay, commutation or extinction angles against time. Since the storage mode is automatically triggered immediately before any major control parameter is changed, a record of the dynamic response to that change is produced. For example the results in fig 5.13 show the response to a step change in current order for varying feedback gains in the current loop. Up to ten individual displays of this type may be saved temporarily for later comparisons, or may be saved permanently on to floppy disks for subsequent hardcopy plotting.

By using the display facilities and the interactive commands, it is possible to interactively modify and analyse the control system, or indeed, any other convertor feature. This ability to interactively develop and analyse any part of the convertors operation and performance should be of great use during development and commissioning of HVDC plant as well as being an extremely powerful research tool. It is a demonstration of the flexibility that may be built into a microprocessor based system.

6.0 Fault Simulation and Detection.

This chapter covers the fault simulation and fault detection sub-systems implemented as part of the HVDC development system. The types of convertor faults that will be considered are first reviewed and, in the light of this, the requirements for the fault simulation and detection sub-systems will be defined. Hardware based simulators are then briefly discussed before the interactive fault simulation sub-system is described in detail.

Following this the requirements for the fault detection sub-system are discussed and existing detection techniques briefly reviewed. The operation of the microprocessor based fault detection sub-system is then presented and finally an application of the fault detection sub-system to fault development control is shown.

6.1 Convertor Faults.

The major part of this chapter deals with the simulation and detection of internal convertor faults. Thus the faults that must be simulated or detected are the commutation failure, the fire through, and the misfire. The backfire, while not entirely unheard of in thyristor valves (Uhlmann, 1975) , (Rumpf and Jarret, 1980), is sufficiently rare that it will not be considered.

6.1.1 Commutation Failure.

This is the most common fault in modern convertors, normally occurring during invertor operation. Commutation failure may be described as the incomplete transfer of dc current from the out-going to the in-coming valve before the commutation voltage reverses polarity and re-forward biases the outgoing valve. The result is that the incoming valve, being reverse biased, switches back off leaving the outgoing valve on, carrying the dc current. The commutation failure results in a considerable disturbance in the dc voltage as a short circuit is established across the dc line for a period during the development of the fault, and this rapidly de-energises the line.

A true commutation failure is not caused by any misoperation of the valves involved ; it is, instead, due to external ac disturbances , or inadequate control of the valve firing times. It may be caused by a commutating voltage reduction , an increase in dc current , late firing of the valves, or any combination of these.

6.1.2 Fire Through.

A fire through may occur in either a rectifier or an invertor, and can be defined as the unscheduled conduction of a valve which has not received a firing pulse. In rectifier operation the fire through causes relatively little disturbance since the convertor is usually operated at low delay angles, but during invertor operation a considerable transient in the dc voltage will result.

The fire through can be caused either by a valve failure produced by transient over voltages, or by a failure in the valve firing circuitry. The fault is unlikely to be permanent if it is due to a transient overvoltage, provided that the valve was not permanently damaged by the resulting over currents. If the fire through was caused by a failure in the firing circuitry, then a permanent fault is likely.

6.1.3 Misfire.

Like the fire through, the misfire can occur in either rectifier or invertor operation. It is usually defined as the failure of a valve to conduct during a scheduled conduction period. In the rectifier mode it may be due to a reduction in the commutating voltage, or may be caused by a control or firing circuit malfunction while in either rectifier or invertor mode. In either mode a large transient in the dc voltage will result.

Both the misfire and the commutation failure are essentially similar in the invertor. During a commutation failure the faulted valve conducts for a short time before turning off unexpectedly, while during a misfire the faulted valve fails to conduct at all. Therefore in this thesis I will refer to faults in which a valve fails to conduct for any reason as a misfire. Faults in which a valve starts conduction, but turns off again shortly after will be refered to as commutation failures. The distinction between the two faults will become more clear later in this chapter.

6.2 Fault Simulation.

6.2.1 Simulator Requirements.

The fundamental requirement of the simulator is that it be able to simulate each of the above faults. However, to be useful the simulator must be capable of much more.

Since any fault may occur singly, periodically, or continuously, the simulator should be capable of simulating multiple as well as single faults. It should also be able to simulate combinations of faults of different types. It must also be easy to use by the operator and, ideally, should be coupled with some form of analysis to permit the operator to analyse the development of, and recovery from, the fault being simulated. Finally it is also desirable that the simulation and analysis take place interactively if the overall system is to be used to its full potential.

6.2.2 Existing Fault Simulators.

Most documented HVDC simulators appear to have little provision for the simulation of convertor faults. It is assumed that manufacturers involved in the design and supply of HVDC systems do have facilities to adequately simulate these convertor faults, but for reasons of their own, they have not seen fit to publish descriptions of such facilities in the technical literature. Those simulators intended to model convertors with mercury arc valves, (Ainsworth and Bowles, 1968), often had some provision for the simulation of arc backs, but later simulators intended to model thyristor valves ,(Machida et al., 1971), appear to have dispensed with even this limited fault simulation capacity. All simulators described appear to have the ability to simulate disturbances in the ac or dc systems and consequently any convertor fault that arises from such a disturbance may be studied, but in general the typical convertor faults described in section 6.1 cannot be adequately studied.

There are of course exceptions to this general trend, and , to the knowledge of the author, complete fault simulation facilities have been included in at least one HVDC convertor simulator (Baldwin, 1972). A simplified form of this simulator has been available for use at the University of Canterbury for several years now, but the discrete logic implementation has proved to be rather unreliable, and is not suitable for interfacing with a microprocessor based system.

6.2.3 Microprocessor Based Fault Simulator.

The fault simulation sub-system was designed to meet the requirements outlined above. Its operation can be split into two distinct parts, the real-time section and an interactive section. The real-time section responds to the commutating voltage zero crossing interrupts and is actually executed on the control processor board. The interactive section communicates with the operator and executes on the interactive processor board.

6.2.3.1 Basic Principles of Operation.

The technique used to simulate all convertor faults is that a fault can be created by suitably manipulating the firing and blocking commands sent to the valve firing circuits. As described in chapter 2, the firing circuit for each valve has two control lines, a fire signal, and a block signal which is capable of overriding the fire command and will prevent a firing pulse from being generated in response to a fire command.

Thus if a block command is sent to a valve before the controller issues a firing command, the valve will be blocked and, as long as the block command is maintained throughout the normal conduction period , the valve will remain blocked and it will appear that a misfire has occurred. If the block command is removed a few degrees before the commutating voltage reverses polarity and the firing pulse to the valve is maintained then a commutation failure will result as there will be insufficient time for the commutation to complete. Similarly if a firing command is sent to a valve with the block command removed, and the valve is forward biased the the valve will start conduction and it will appear that a fire through has occurred.

Thus any convertor fault can be simulated by suitably manipulating the block and fire commands to each valve at each zero crossing during the period when simulation is requested.

Faults in the ac or dc systems may also be simulated to examine their effects on the convertor. This has been implemented by using the microprocessor to close a set of relay contacts at a specified time and to re-open them a fixed time later. In this way both single and multiple phase to phase or phase to neutral ac faults, as well as dc line to line or line to ground faults may be simulated. Both the starting time and the duration of the fault are specified by the operator, and the short circuit may be imposed for durations of up to 1000 cycles.

6.2.3.2 Real Time Software.

This section of code corresponds to the interrupt service routine that responds to the commutating voltage zero crossing interrupts, and to two timer interrupt service routines. The actual sequences of firing and blocking commands are setup in a series of arrays by the interactive routines in response to the operators fault specification. At each interrupt the service routine steps through the command arrays, outputing a new set of fire and block commands to the firing circuitry. Fig 6.1 shows the structure of the zero crossing interrupt service routine.

Refering to fig 6.1 the first step is to determine the crossing number and save it for later use. Following this a flag is tested to see if simulation is enabled, and if so then the pointer into the command arrays is recalled and used to output the next fire and block commands to the firing circuitry in the convertor. If an ac fault is required then the command to close or reopen the relay will be issued. A pair of timers are then started, the first runs for a period set by the operator to provide an adjustable delay until a fire through is initiated. The second timer runs for sixty degrees minus the extinction angle required to create the commutation failure and permits the operator to adjust this angle. Both of these timers will generate an interrupt on reaching terminal count. Once the timers have been started the command array pointer is incremented and the value specified for the next block command is examined. If this is set to the end of simulation value then subsequent simulation is disabled by clearing the simulation enabled flag and the array pointer is reset, ready for the next simulation.

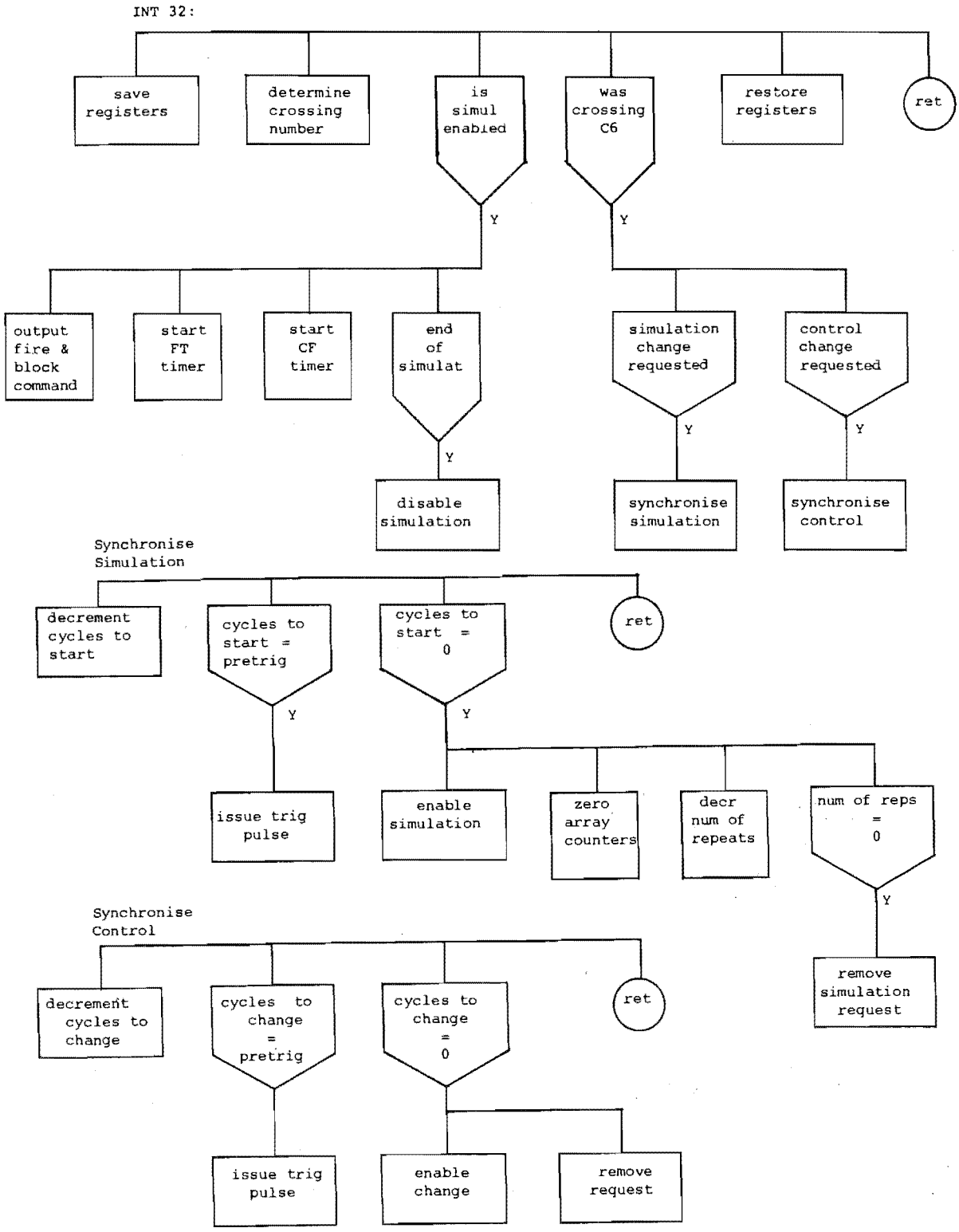


Figure 6.1 Zero Crossing Interrupt Routine

Alternatively, if simulation was not enabled then any existing blocking command is removed and neither timer is started. The remaining section of code is executed when the operator has requested a fault to be simulated and simulation is not currently enabled. This permits the actual start of simulation to be synchronised with the convertors operation.

If simulation has been requested and is not enabled then at every C6 crossing interrupt a counter is decremented, the initial value of which was set by the operator. This counter provides a short delay between the instants when simulation is requested and when simulation is enabled. This has two advantages, firstly it provides time for the operator to reset any external equipment such as storage oscilloscopes or chart recorders, and secondly it provides a means of synchronising the starting of the recording devices and the start of simulation with the convertor. When the count reaches a pre trigger value, also specified by the operator, a trigger pulse is issued to any external data acquisition equipment, and at the same time the internal data acquisition routines are toggled to the storage mode. This typically occurs two or three cycles before the fault is initiated and permits the leading edge of the fault to be captured and analysed. When the count finally reaches zero then the array counters are reset to the start of the command arrays and the simulation enabled flag is set. Using this technique ensures that simulation always starts at the first crossing following the C6 crossing.

If the simulation was started during the current execution of the routine then the number of repetitions requested by the operator is decremented, and if it is now zero, then the simulation request is removed. This allows a particular fault sequence to be repeated a specified number of times.

In order to obtain sufficient data for analysis of the control sub-system, it is necessary to trigger any recording devices and to set the storage mode flag before any change in the control sub-system occurs. Thus at every C6 crossing, a similar process to that described for a simulation request is also used for a control change request.

The remainder of the real time routine comprises the interrupt service routines for the two timers. The structure of these is shown in figs 6.2 and 6.3.

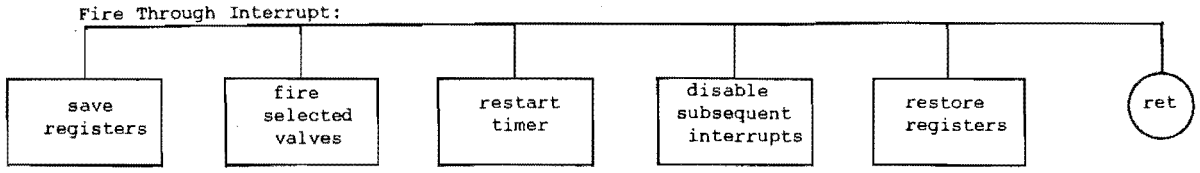


Figure 6.2 Commutation Failure Timer Routine

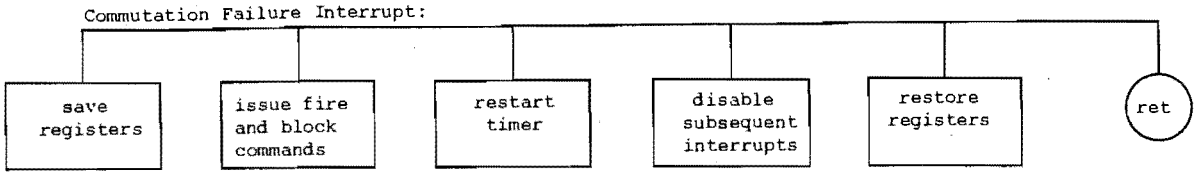


Figure 6.3 Fire Through Timer Routine

The operation of both is essentially similar in that each uses its own array pointer to index into either the fire through or commutation failure command arrays and output the appropriate commands to the fire and block ports. The fire through routine merely fires the appropriate valve, while the commutation failure routine deblocks and fires the incoming valve, which should be receiving firing pulses from the controller by now anyway. It also issues a firing pulse to the outgoing valve, ensuring that it will resume conduction as soon as the commutating voltage reverses polarity, thus producing the required commutation failure.

6.2.3.3 Interactive Software.

This in turn can be divided into two parts, the first dealing with the specification and initiation of the desired fault, while the second deals with the analysis of the recorded results.

6.2.3.3.1 Interactive Fault Specification.

These routines permit the operator to enter the types of faults to be simulated as well as to change any simulation parameter such as the fire through or commutation failure delays, the start of simulation delay or the time between triggering the data acquisition devices and when simulation commences.

Currently the fault simulation software permits the specification of multiple faults of mixed types over a four cycle period. This basic pattern can then be automatically repeated up to 255 times with a pause of up to 255 cycles between repetitions. Both the number of repetitions and the pause in between may be specified by the operator as part of the fault specification procedure. This effectively permits faults of up to 1024 cycles duration to be simulated.

When a set fault command is issued, the existing fault specification is cleared and the operator is then prompted to select which of the four cycles the fault should occur in. The type of fault and the valve in which it is to occur are then requested. Once the desired combination of faults has been entered, the operator is then prompted to specify the combination as either being a single fault or a multiple fault. If a multiple fault has been requested then the number of repetitions and the interval between each repetition are prompted for. Once a fault combination has been entered it may be initiated by entering the appropriate command. To save the operator from having to re-enter parameters that may not change from one run to the next, the system automatically enters certain default values, such as fire through delay or pre trigger time. These may be changed at any time by entering the appropriate command. In this manner the operator is able to quickly specify and simulate selected faults.

6.2.3.3.2 Interactive Fault Analysis.

The simulation of a fault automatically triggers the data acquisition storage mode. This provides detailed data on the convertor's operation for the period from immediately before until about ten cycles after the fault was initiated. This data may be analysed in many ways.

The first technique developed was to examine the graphs of delay, commutation and extinction angles recorded during the fault in much the same way that the control sub-system performance was assessed. This is useful in determining transient behaviour during, and speed of recovery from, a fault. However, the occurrence of the fault disrupts to a small extent the measurement process and this limits the usefulness of this technique. It also provides little information on the exact development and recovery of the fault. A second display mode was introduced to permit

this information to be readily obtained in terms of which valves actually failed and what type of faults occurred.

The exact turn-on and turn-off times of each valve are available in the data base and this information may be processed to provide an exact record of the conduction history of each valve in the bridge. Measurements are made to the nearest hundredth of a degree, but the conduction patterns are only calculated to the nearest degree to reduce the data involved to a manageable size. These conduction patterns may then be presented to the operator in the form of bar-graphs, providing an easily interpreted history of the fault.

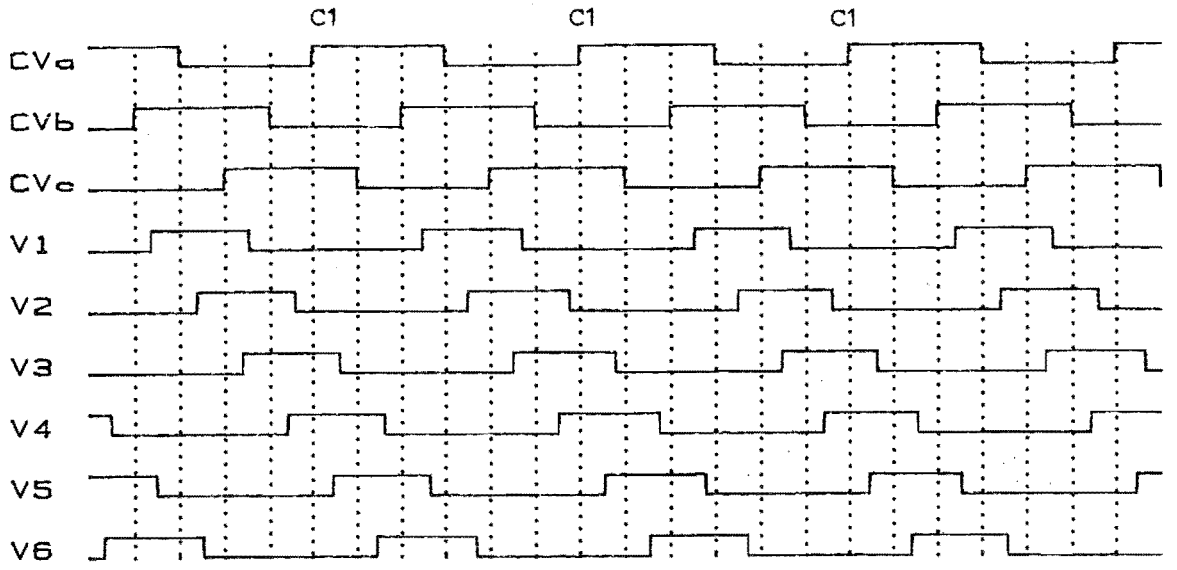


Figure 6.4 Normal Invertor operation

Fig 6.4 shows a typical display obtained in this manner for normal invertor operation. The three commutating voltage waveforms have been included to permit the zero crossings to be used as timing references. Appendix 1 contains a definition of the numbering convention used in these diagrams. In figure 6.4 the sequential nature of the valve conduction patterns during normal operation is readily apparent. The display on the operators screen is only a small portion of the total record of the fault development. The type of terminal used in the project is limited to displaying only 80 characters on a line. This causes a trade off between the resolution of the display and the amount of the fault pattern that may

be displayed on the screen at any one time. The default step size on the screen is fifteen degrees, which means that approximately three cycles of the fault are visible. If a finer resolution is required then step sizes of down to one degree are available, although correspondingly less of the overall pattern is then visible. To allow the operator to examine the entire pattern in detail the starting position of the display in the data is movable. Thus the operator can select the starting position and resolution to obtain the most appropriate display for his needs.

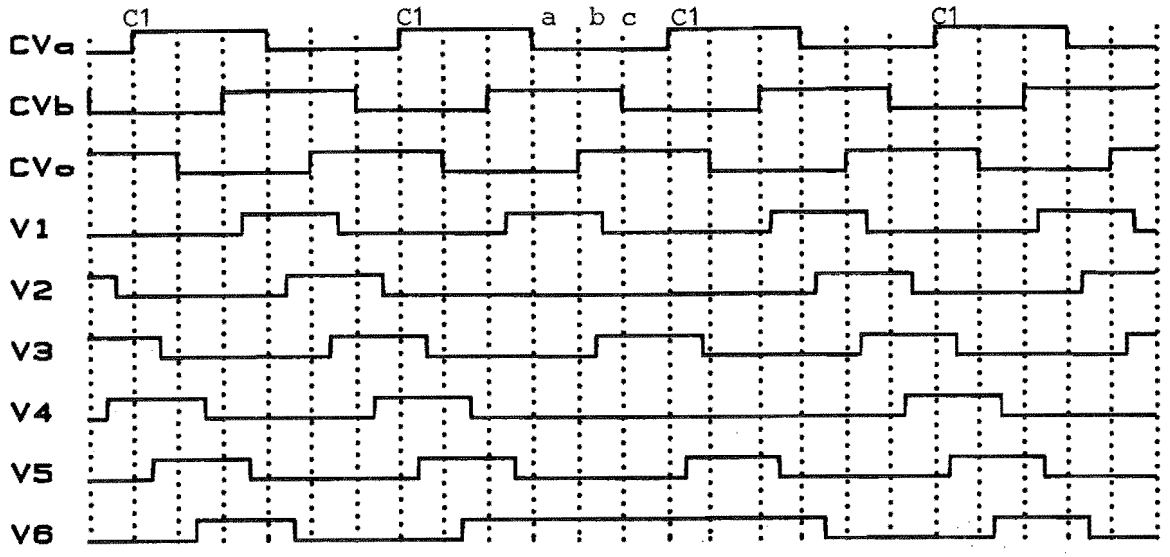


Figure 6.5 Uncontrolled Misfire
Strong ac System

The same display format is used in fig 6.5 which shows the response to the simulation of a single misfire in valve V2. It can be seen that at point A valve V2 has not turned on, which leaves valve V6 on. At point B valve V3 turns on establishing a short circuit across the dc line and de-energising it. Note that at point C valve V4 also fails to conduct. This is because the commutating voltage is now the yellow-red phase to phase voltage rather than the usual blue-red . Therefore, if the commutation is to take place, then valve V4 must be fired before crossing C6, as this is when the yellow-red voltage will reverse polarity. Due to the equidistant nature of the controller, this will not occur, and consequently by the time valve V4 is fired by the controller it is no longer forward biased. A misfire of valve V4 therefore results, and valve

V6 continues in conduction. Eventually the convertor recovers with valve V2 conducting normally in the subsequent cycles.

The bar graph form of analysis permits full interactive diagnosis of convertor faults, and will later be used to develop a fault detection algorithm. Coupled with the interactive fault specification routines it forms an extremely useful development and diagnostic aid. Further results obtained using these techniques will be presented in chapter 7.

6.3 Fault Detection.

Detection of faults in the ac or dc systems is adequately handled by existing techniques. The detection of internal convertor faults can be improved substantially, and the rest of this chapter concentrates on this area.

6.3.1 Existing Convertor Fault Detection Techniques.

Existing fault detection techniques (Uhlmann, 1975), rely on detecting the overcurrents that result from the convertor fault. Thus, for example, a commutation failure is detected by detecting the difference between the rectified ac currents and the dc current. This type of method, while relatively simple to implement suffers from the disadvantages that it is both slow, the effect is being detected rather than the cause, and it produces rather ambiguous results. That is there is no indication of the valve in which the fault occurred, and it is difficult to differentiate between a commutation failure and, for example, a misfire.

6.3.2 Direct Digital Fault Detection.

As was the case in the control system with proposals for direct digital control, so the advent of digital logic families in the 1960s lead to proposals for direct digital fault detection schemes (Arrillaga and Galanos, 1970a) (Reeve, 1967a,1967b). Several authors demonstrated that it was possible to make an exhaustive classification of the logical behaviour of the valve conduction states during all fault conditions. Based on these logic conditions and coupled with the knowledge of the commutating voltages and the firing commands, it was possible to construct hardware based on discrete logic circuits to perform the fault detection task. These schemes

have apparently not met with commercial approval, despite the considerably improved detection times possible, and the major reason for this is probably the excessively complicated circuits required and doubts over the resulting reliability.

6.3.3 Microprocessor Based Fault Detection Scheme.

6.3.3.1 Theory of Operation.

Fig 6.4 showed the conduction pattern for a normally operating convertor. The standard valve conduction sequence is readily apparent in this, with the valve turn-ons and turn-offs following in order. Similar conduction patterns have been recorded for the various convertor faults, and these are shown in figs 6.6 to 6.8.

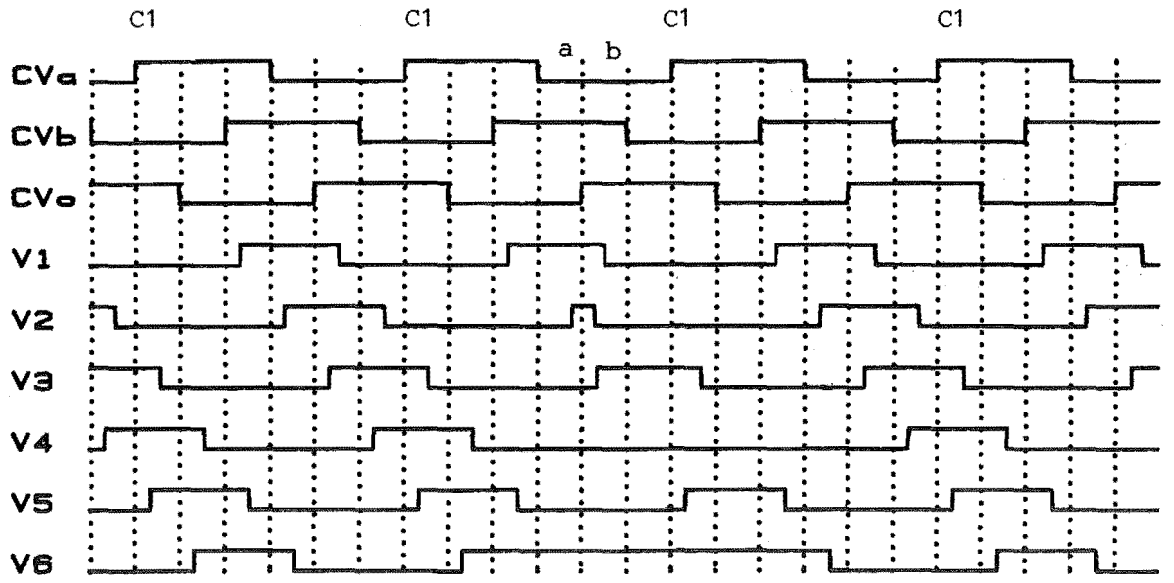


Figure 6.6 Uncontrolled Commutation Failure
Strong ac System

The conduction pattern of a commutation failure involving valves V2 and V6 is shown in figure 6.6. Valve V2 is fired by the controller at point A, and it starts conduction. Unfortunately there is insufficient time for the current in valve V6 to decay before valve V6 is re forward biased again at crossing C5. Valve V6 therefore takes over conduction from valve V2 again and valve V2, now reverse biased, consequently turns off.

This leaves valve V6 in conduction, and the next valve in that side of the bridge to be fired, valve V4, now has as a commutating voltage the yellow-red phase-phase voltage rather than the usual blue-red. If valve V4 is to turn on when fired and commutate off valve V6, it therefore must be fired before crossing C6. As shown in figure 6.6 this does not happen and valve V4 experiences a consequential misfire.

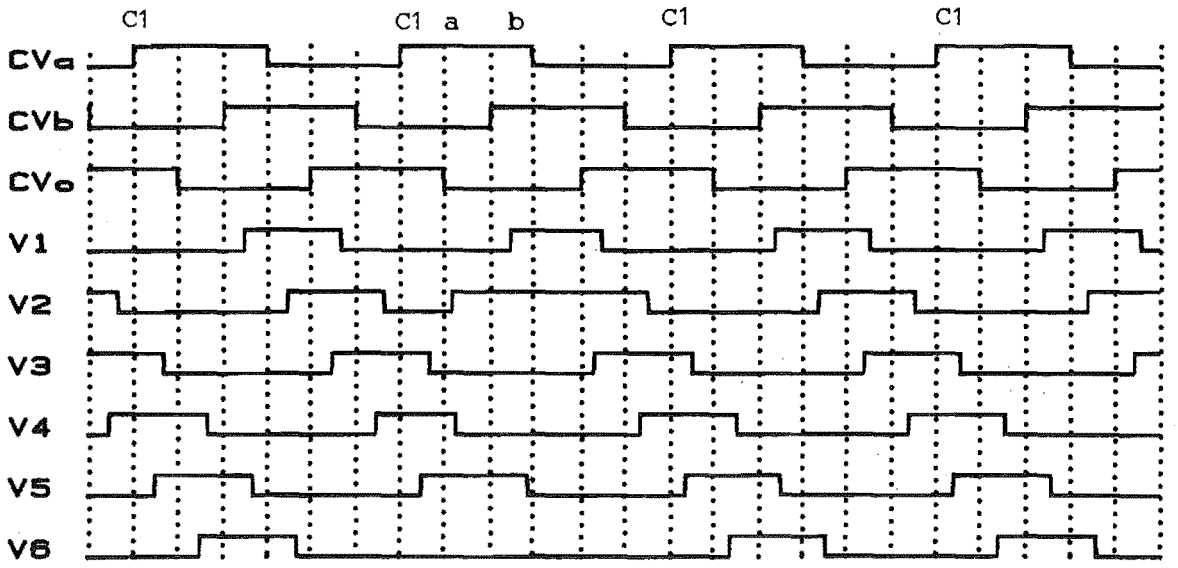


Figure 6.7 Uncontrolled Fire Through Strong ac System

Figure 6.7 shows the conduction pattern for a single fire through of valve V2. In this example valve V2 has fired through about ten degrees after it was first forward biased. Valve V4 is commutated off, resulting in a shorter than normal on time. The commutating voltage across valve V6 is now the blue-yellow phase-phase voltage rather than the red-yellow that would normally be present. This implies that if valve V6 is to successfully start conduction then it must be fired before crossing C2. This does not occur, and consequently when valve V6 is eventually fired it is reverse biased and a misfire results.

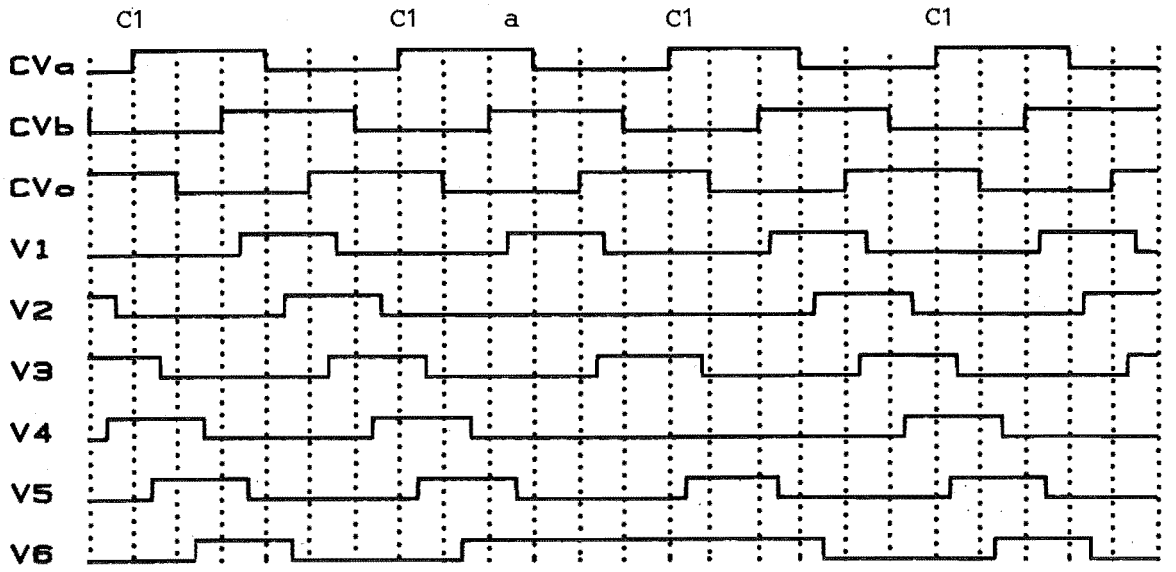


Figure 6.8 Uncontrolled Misfire
Strong ac System

The development of a misfire is shown in figure 6.8. The lack of conduction in valve V2 is readily apparent. Valve V6 remains in conduction because no valve has yet been fired to commutate it off and this results in the commutating voltage across valve V4 being the yellow-red phase-phase voltage rather than the blue-red . This implies that if valve V4 is to successfully commutate off valve V6 then valve V4 must be fired before crossing C6, rather than crossing C1 as would normally be the case. Unfortunately the equidistant controller will not fire valve V4 until well after the C6 crossing and consequently valve V4 misfires when it is eventually fired.

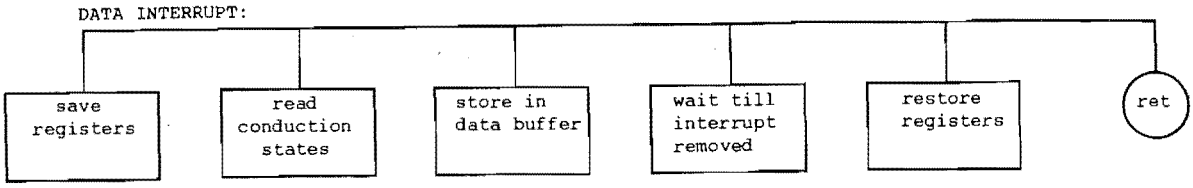
Examining the patterns of figure 6.8 in more detail, it can be seen that up until point A the convertor was operating normally. Each valve conducted in sequence and it is possible to accurately predict the next expected valve turn-on and turn-off. However, following the turn-on of valve V1 and the related turn-off of valve V5, instead of the next detected turn-on being in valve V2 it is in fact in valve V3. This unexpected turn-on can be used to indicate that a fault has occurred. The type of fault cannot be directly determined from the unexpected turn-on because the turn-on could have resulted from either a fire through of valve V3 or a misfire of valve V2. It is therefore necessary to perform an additional test to distinguish between the two possibilities. The misfire was defined as being the total lack of conduction in a valve despite the fact that a firing command was issued to it. Therefore a misfire can be detected if, on detection of an unexpected valve turn-on, the valve that was actually expected to turn-on is checked to see whether a firing command was issued to it and whether it is in conduction. Similarly a fire through was defined as the conduction of a valve without that valve having received a firing pulse. A fire through can be detected therefore by checking that no firing pulse was issued to a valve when it unexpectedly begins conduction.

A commutation failure can be detected using similar techniques. As shown in figure 6.6, the commutation failure in valve V2 is characterised by the unexpected turn-off of valve V2. Under normal operation, following the turn-on of valve V2 at point A, the next expected event would be the turn-off of valve V6. Instead the turn-off of valve V2 is recorded. That a commutation failure has occurred can be confirmed by examining the firing pulse to valve V2. If it is still present then valve V2 has undergone a commutation failure.

By monitoring the valve turn-on and turn-offs and performing a few simple tests in the event of an unexpected occurrence it is possible to detect all of the typical convertor faults.

6.3.3.2 Microprocessor Based Implementation.

Interrupts are generated by the convertor at each valve turn-on and turn-off. These are already supplied to the data acquisition processor where they are used by the data acquisition sub-system to measure the valve's delay, commutation and extinction angles. To implement the fault detection sub-system however, an extra processor is required as the response time of a single processor handling both tasks would be too slow. The on/off interrupts are therefore also supplied to the fault detection processor, and the interrupt service routine that responds to them is shown in fig 6.9 .



Fault Detection Data Interrupt Routine

Figure 6.9

Since, especially during fault conditions, the intervals between valve switching can be extremely short, the interrupt service routine merely places the new conduction pattern in to a data buffer. This minimises the time spent in the service routine and allows the detection of a turn-off that closely follows a turn-on which would otherwise be missed. The data stack entries are then cleared by the fault detection mainline routine, shown in fig 6.10a .

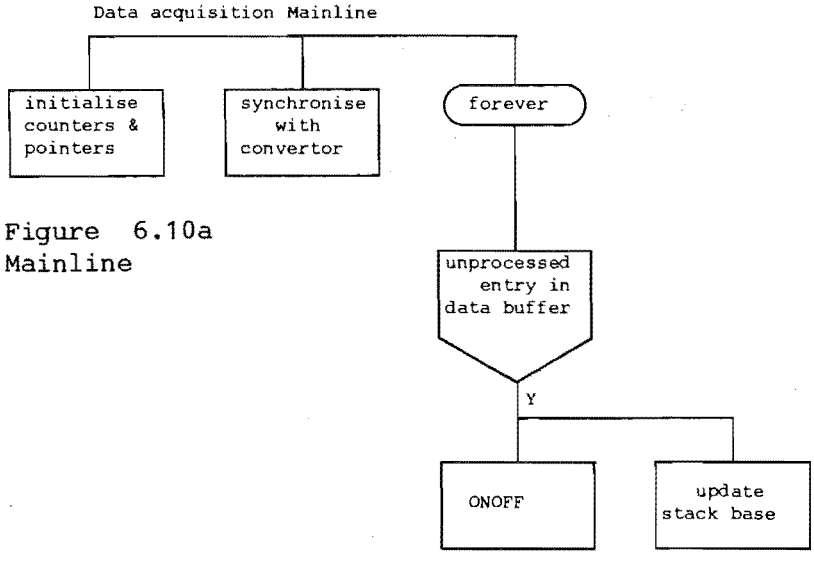


Figure 6.10a
Mainline

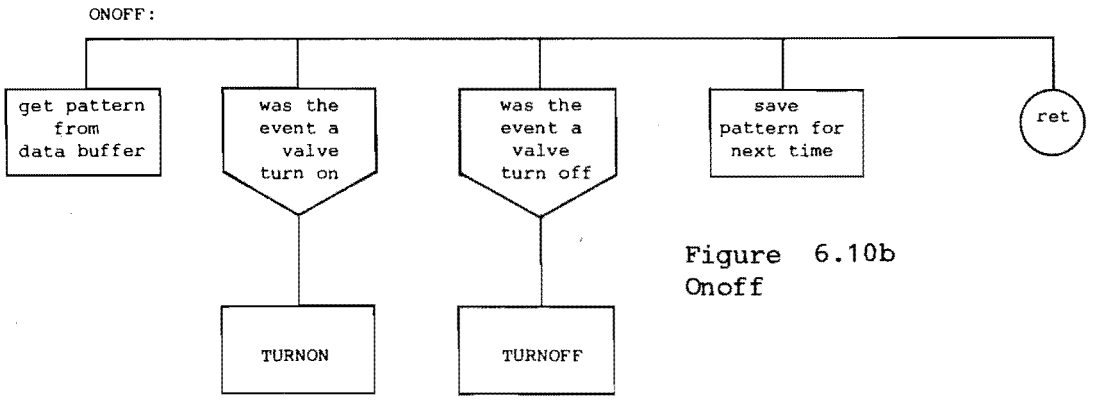
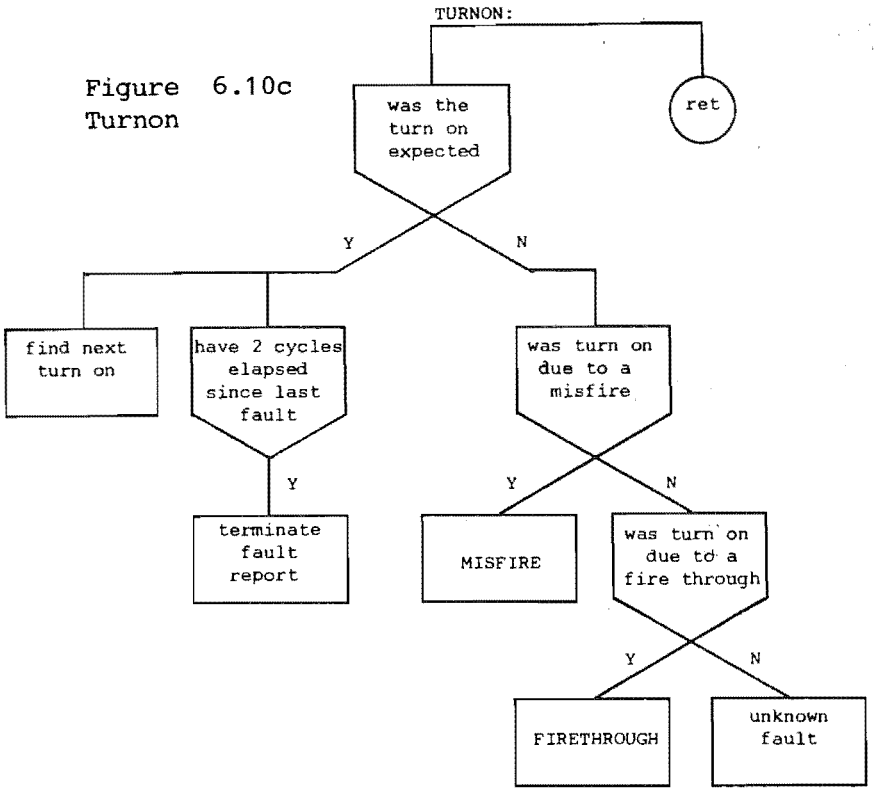


Figure 6.10b
Onoff

The fault detection mainline operates by continually looping, looking for any unprocessed entries in the data buffer. Each entry is cleared by routine ONOFF, shown in figure 6.10b, and the buffer base is then incremented. By exclusive ORing the new conduction state with the previous pattern it is possible to determine which valves have turned on or off.

Figure 6.10c
Turnon



If a valve turn-on is detected then routine TURNON, figure 6.10c, is called. In this routine the recorded turn-on is compared with the expected turn-on. If the two are the same then the convertor is operating normally and the next expected turn-on is calculated. A record of time since the last fault occurred is also maintained, so this is incremented if the convertor is operating normally, or is reset to zero if a fault has been detected. The time since the last fault is used to decide whether a fault is a consequence of an earlier fault, or whether it is a new fault. It is assumed that any fault occurring within two cycles of a previous fault is a consequence of that earlier fault.

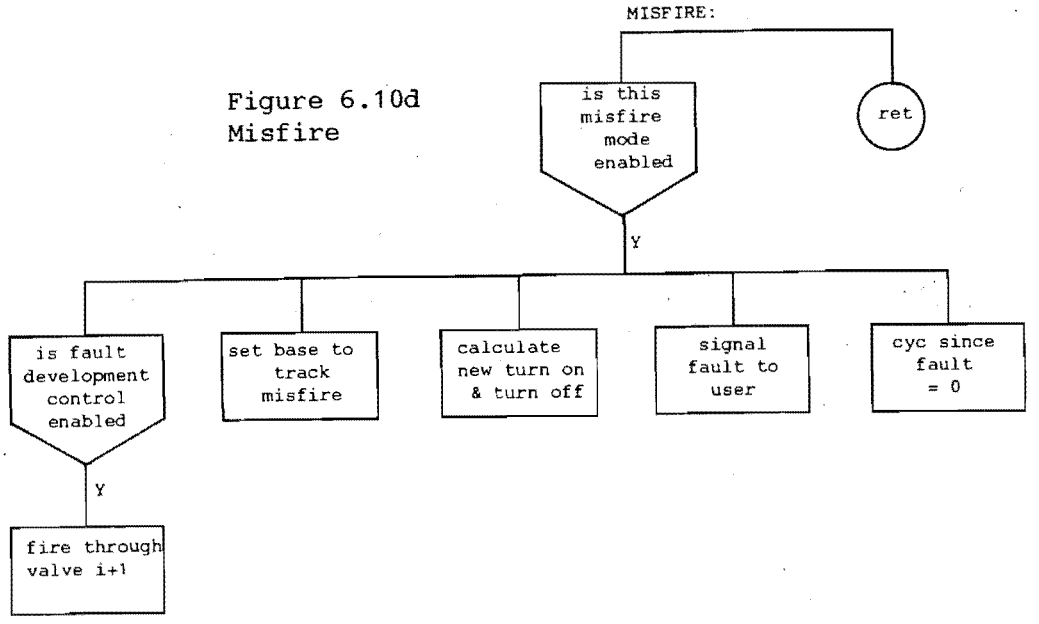


Figure 6.10d
Misfire

If the recorded turn-on did not match the expected turn-on then the occurrence of a misfire or a fire through is tested for, figs 6.10d, 6.10e. A misfire is deemed to have occurred if the valve that was expected to have turned on is not in conduction at the unexpected turn-on of another valve despite its having received a firing pulse from the controller. A fire through will be detected if the valve that has unexpectedly turned on has not received a firing pulse from the controller. If the unexpected turn-on does not meet either of these conditions then it is ignored.

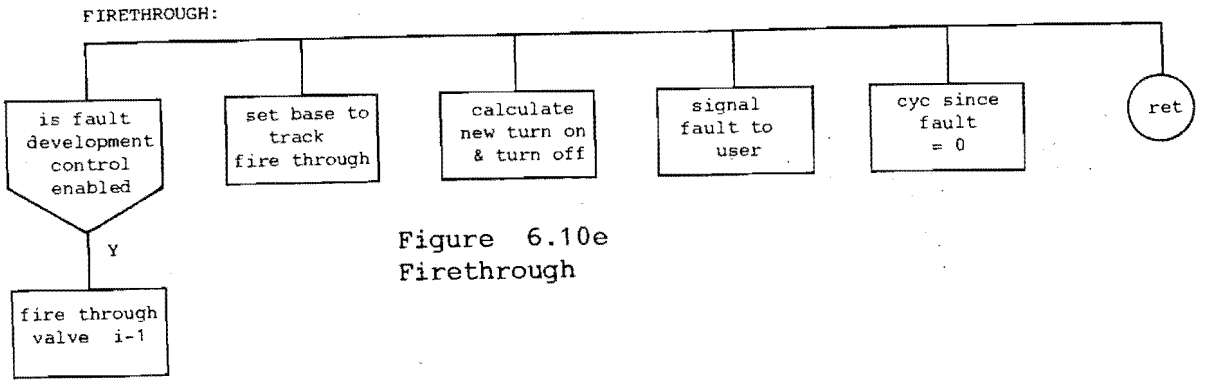


Figure 6.10e
Firethrough

If a fault has been located then the type of fault and the valve in which it occurred are placed in to buffer from which they will eventually be processed by the interactive processor and reported to the operator. Once the fault has been reported the new values of expected turn-on and turn-off are calculated to take account of the fault that has just occurred.

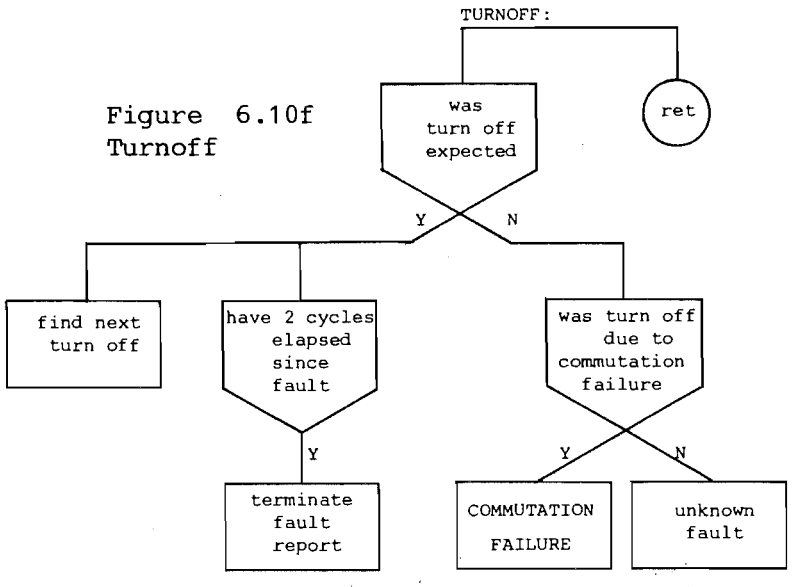


Figure 6.10f
Turnoff

After processing any turn-on that occurred the ONOFF routine then tests for any valve turn-offs that may have occurred. If a turn-off has been detected then it is processed, by routine TURNOFF, figure 6.10f in the same way as a turn-on except that a commutation failure will be tested for in the event of an unexpected turn-off. A commutation failure will be detected by routine COMMUTATION FAILURE, fig 6.10g, if the valve that has just turned off is still receiving a firing pulse from the controller.

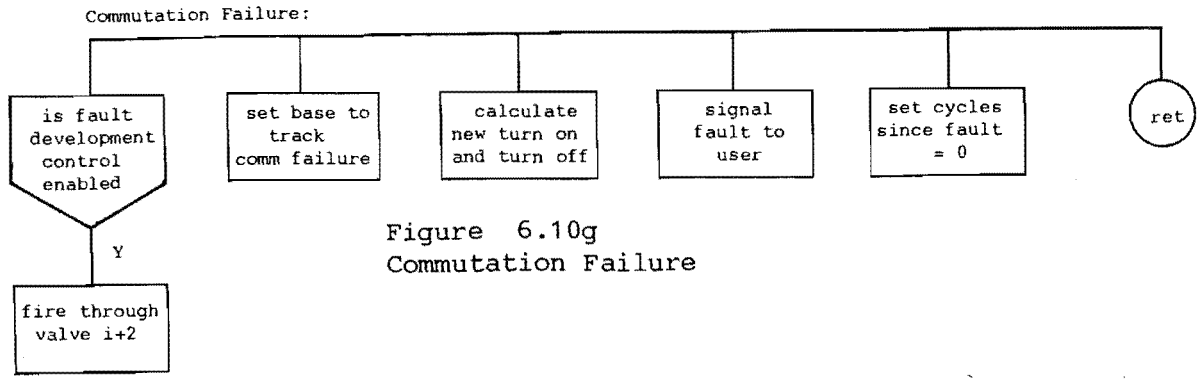


Figure 6.10g
Commutation Failure

Again the fault is reported and new expected turn-on and turn-offs calculated. As in the case of the turn-on, if the turn-off was expected then the next expected turn-off is calculated and the time interval since the last fault is updated. The final task for the ONOFF routine is to save the current valve conduction pattern for use next time it is called.

Using this technique it is possible to produce an indication of the fault within five degrees of the unexpected turn-on or turn-off occurring. This implies that for a commutation failure or a fire through it is possible to detect the fault within five degrees of the start of the fault. For a misfire however a longer delay exists. Using the method described above the first point at which the lack of conduction in valve Vi can be detected is at the subsequent turn-on of valve Vi+1, nominally sixty degrees after valve Vi should have started conduction. Thus a delay of sixty five degrees from start of fault to detection is typical for a misfire. This long delay will be shown to have important consequences in the section on fault development control.

An alternative technique to detect a misfire is for the controller to start a timer which runs for a period of 10-15 degrees. The timer is restarted each time the controller issues a firing pulse. On an interrupt being generated at the timer's terminal count the fault detection system can then test to see whether the valve just fired has in fact started conduction. This method implies that a valve that has not started conduction within 10-15 degrees of being fired will fail to conduct for the

remainder of its normal conduction period. This is not likely to be a problem in practice as the minimum delay angle loop in the controller should ensure that the valve is sufficiently forward biased to take over conduction as soon as it is fired. The interrupt service routine associated with this technique is shown in fig 6.11.

Refering to fig 6.11, the routine operates by first checking that this method has in fact been enabled. This is necessary because both forms of misfire detection have been included in the HVDC development system to allow the operator the flexibility to choose the most appropriate routine at any time. If the alternative routine is enabled it then checks that the valve just fired by the controller has actually started conduction. If the valve has failed to start conduction then the fault is reported to the interactive processor in the same manner as previously described, and new expected turn-on and turn-offs are calculated. If the valve has in fact turned on then no further action is taken as the turn on will have been picked up and processed by the normal fault detection routines.

Both techniques have been investigated and the consequences of using either will be explained in the section on fault development control.

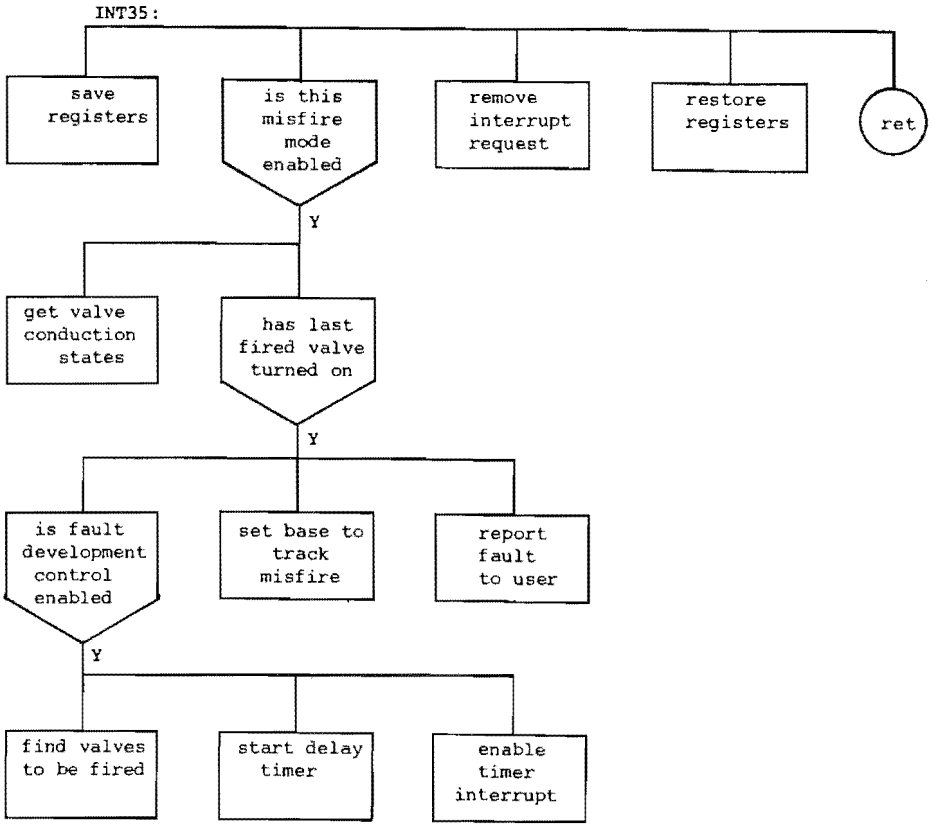


Figure 6.11 Alternative Misfire Detection Routine

6.3.3.3 Tracking Fault Developments.

The fault detection technique used requires that the next expected valve turn-on and turn-off can be reliably predicted. During normal operation this is not a problem as the valves conduct in a cyclic order with valve Vi+1 always following valve Vi. During faults however the turn-on and turn-off sequences change and it is necessary to track the changes if the faults are to be correctly identified. For example, consider the development of a fire through of valve V2 as shown in fig 6.7. At point A valve V2 has fired through and the turn on sequence recorded during the development of the fault is

on4 - on5 - on2 - on1 - on3 - on4 - on5 - on6 - ...
↑fault

Note that valve V6 also misfired as a result of the fire through of valve V2. It is apparent from this that during a fault the next expected valve cannot be calculated as just being the current valve + 1. To overcome this problem a pair of offset arrays were created, one for the valve turn-ons and the other for the valve turn-offs. These contain the number that must be added to the current valve to obtain the correct next expected valve. Entries in the arrays are made for normal operation and for the various expected fault types. Each array is addressed by a pair of pointers, a base pointer and an index pointer. The base pointer determines which area of the array is being used, while the index pointer cycles from 0 to 5. Thus the base pointer is normally set to zero to track normal operation of the convertor. When a fault is detected however the value of the base pointer is changed to track the type of fault just identified.

Refering back to the fire through in fig 6.7 and ignoring the valve turn-off sequences for the moment, once the fault is identified at point A the expected valve turn-on sequence for the fault would be

on2 - on6 - on1 - on2 - on3 - on4 - on5 - on6 - ...
fault

or by expressing the fault as the difference between successive valves the sequence is identified as

+4, +1, +1, +1, +1, +1.

Combining this with the simple pattern for normal operation results in the offset array shown in table 6.1.

+1, +1, +1, +1, +1, +1,
+4, +1, +1, +1, +1, +1,

During normal operation the base pointer has been set to zero and the index pointer cycles along the first line of the table. Upon detection of the fault at point A in fig 6.7, the base pointer is set to six and the index pointer is reset to zero. Thus the next turn-on is calculated as being expected in valve V2+4 or valve V6. In fact this does not occur, as valve V6 has suffered a consequential misfire. It is necessary to modify the simple offset array in table 6.1 to take account of this.
Table 6.2 shows the additions necessary to track a misfire as well as a fire through.

+1, +1, +1, +1, +1, +1,
+4, +1, +1, +1, +1, +1,
+1, +1, +1, +1, +1, +1,

Note that the contents of the misfire row are the same as those for normal operation for the valve turn-on table. This is not the case in the valve turn-off table, and as the same pointers are used in both tables, separate entries are required in each table.

Thus following the fire through of valve V2 at point A in fig 6.7, a misfire in valve V6 will be detected at point B. The misfire is reported and identified as being a consequence of the earlier fire through. The base pointer is then set to twelve and the index pointer is again reset to zero. The next turn-on is then calculated as being expected in valve V2. In fact the next turn-on that will be detected is that of valve V3, as valve V2 is already on due to the earlier fire through. A misfire of valve V2 is not detected however because the conditions for a misfire require that the valve not be conducting. Instead the unexpected turn-on is ignored, as it does not match any of the set fault detection criteria, and with the base pointer remaining set at twelve, the index pointer is incremented and the next turn-on expected is calculated to be in valve V4. This eventually occurs, and the fault is tracked through the remainder of its development.

When the index pointer eventually reaches the end of a particular line in the table it is reset back to the start and the base pointer is set back to the first line of the table and the fault detection sub-system will track normal operation until the next fault occurs. Using this method

permits the successful identification and tracking of complex series of convertor faults. The total offset table is not large with normal operation, fire throughs, and commutation failures only requiring a one line entry, while in practise a two line entry is required for misfires. The amount of memory required for this table is a small overhead for the advantages provided by the technique.

6.4 Fault Development Control.

6.4.1 Principles of Operation.

This refers to a technique (Arrillaga and Galanos, 1969) in which the normal development of a fault can be altered both to reduce the disturbance that the fault would otherwise cause, and to increase the probability of the convertor recovering from the fault without blocking and bypassing being necessary . This method is only feasible if a fast fault detection scheme which is capable of providing unambiguous indication of the type and location of the fault is available. When it was first suggested in the literature a practical objection was raised. This related to the commutation time required during fault conditions.

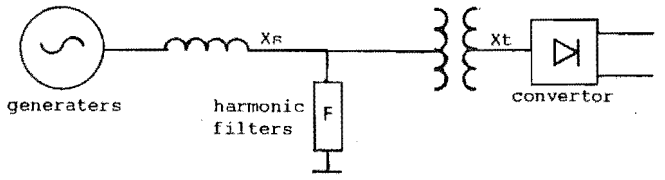


Figure 6.12 Convertor Line Diagram

A typical HVDC system is shown in a line diagram in fig 6.12 . The voltage at the convertor bus is, in the presence of ac harmonic filters, normally sinusoidal. Thus the commutating reactance is limited to the reactance of the convertor transformer (Kimbark, 1971), (Arrillaga, 1983). Under steady state conditions, therefore, the commutation time is determined by this transformer reactance and the magnitude of the current being changed between the valves. During a fault however the voltage at the convertor bus is likely to be badly distorted, particularly on the weak systems typical of modern HVDC links. The result is that now the

commutation reactance can no longer be considered to be just the transformer reactance, and the ac system reactance must also be taken in to account . Thus the commutation time during a fault is considerably extended by both the increased reactance and the increased current flowing as a result of that fault. This increase in commutation time, it was suggested, would prevent the techniques proposed by Arrillaga and Galanos from working successfully, and no trials involving physical convertors were never carried out. Computer based simulations of the technique have been performed, (Arrillaga and Giesner, 1972), (Galanos et al., 1984), and these indicate that the technique should be successful on strong systems, but that problems may be experienced on weaker systems.

The basic technique of fault development control is to prevent the unintentional de-energisation of the dc line during a fault. This has the effect of reducing the fault current that flows, and consequently reducing any possible damage to the valves, as well as increasing the probability of a natural recovery from a fault without deliberate bypassing being necessary. The availability of the fast fault detection system described above together with a convertor model capable of simulating a range of short circuit ratios made the implementation and evaluation of a fault development control scheme possible.

6.4.2 Implementation.

The fault development control works closely in conjunction with the fault detection routine, and in fact forms an operator selectable subroutine. Thus when a convertor fault has been detected, a development control flag is tested and, if the flag is set, the necessary corrective action is taken. The basic technique is to remove the valve that should not be conducting and replace it with a valve that restores the normal operating sequence.

For example in fig 6.6 a commutation failure is detected shortly after point A. At this point valve V6 which should have turned off has in fact remained on, and at point b following the normal turn-on of valve V3, a short circuit will be established across the dc line. Valve V2 is by now reverse biased and so would not conduct if refired but valve V4 is forward biased and should commutate valve V6 off if fired. The effect of doing this is shown in the conduction pattern of fig 6.13.

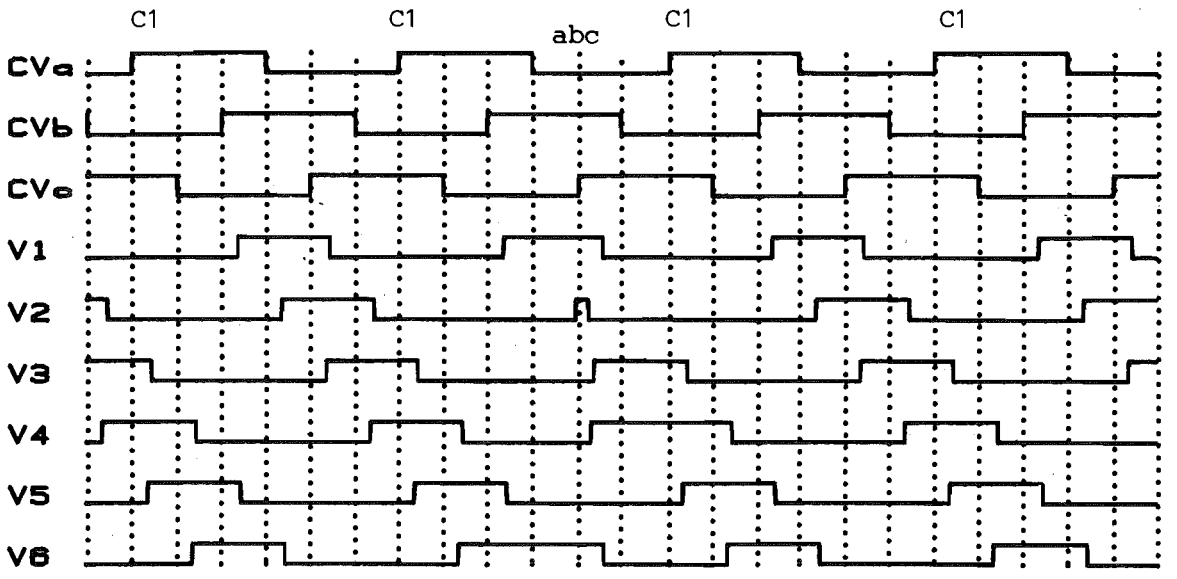
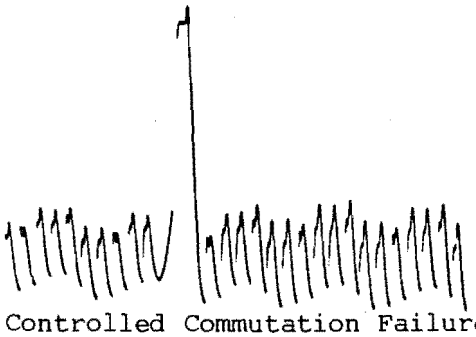


Figure 6.13 Controlled Commutation Failure
Strong ac System

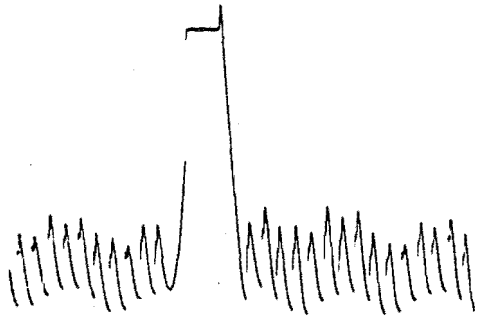
For clarity a large (20:1) ac system short circuit ratio was used to obtain these results in order to demonstrate the basic principles of operation. In fig 6.13 it can be seen that following the detection of the commutation failure at point A, valves V3 and V4 were fired through at point B and valves V1 and V6 finally commutate off at point C. Normal operation is restored when valve V5 is eventually fired by the controller. Fig 6.14 shows the disturbance in the invertor voltage waveform that results. For comparison fig 6.15 shows the same waveforms for an uncontrolled commutation failure.

It is necessary to fire through both V3 and V4 since if V4 only was fired through then a short circuit between V1 and V4 would be established until V3 was fired by the controller. Further results demonstrating the effect of lower short circuit ratios on the fault development control process will be presented in chapter 7 .



Controlled Commutation Failure

Figure 6.14



Uncontrolled Commutation Failure

Figure 6.15

Similar techniques can be used for both fire throughs and misfires. In fig 6.16 the short circuit is established as soon as valve V1 fires through since valve V4 is already on. The short circuit may be removed by firing through valve V6 to commutate off valve V4, and normal operation will be restored with the eventual firing of valve V2 by the controller. Figure 6.17 shows the dc waveform for the controlled development, while figure fig 6.18 shows the dc waveform for an uncontrolled fire through.

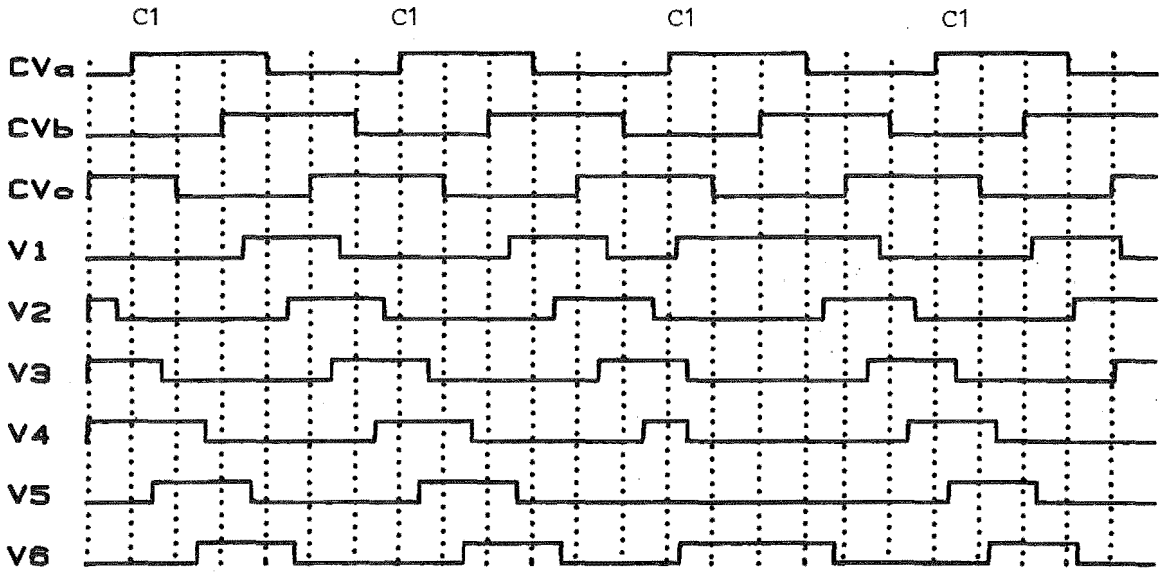
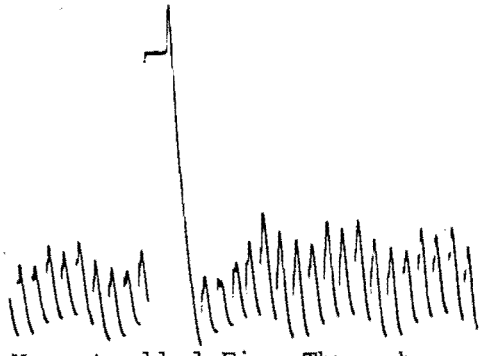


Figure 6.16 Controlled Fire Through Strong ac System



Uncontrolled Fire Through

Figure 6.18

Controlled Firethrough

Figure 6.17

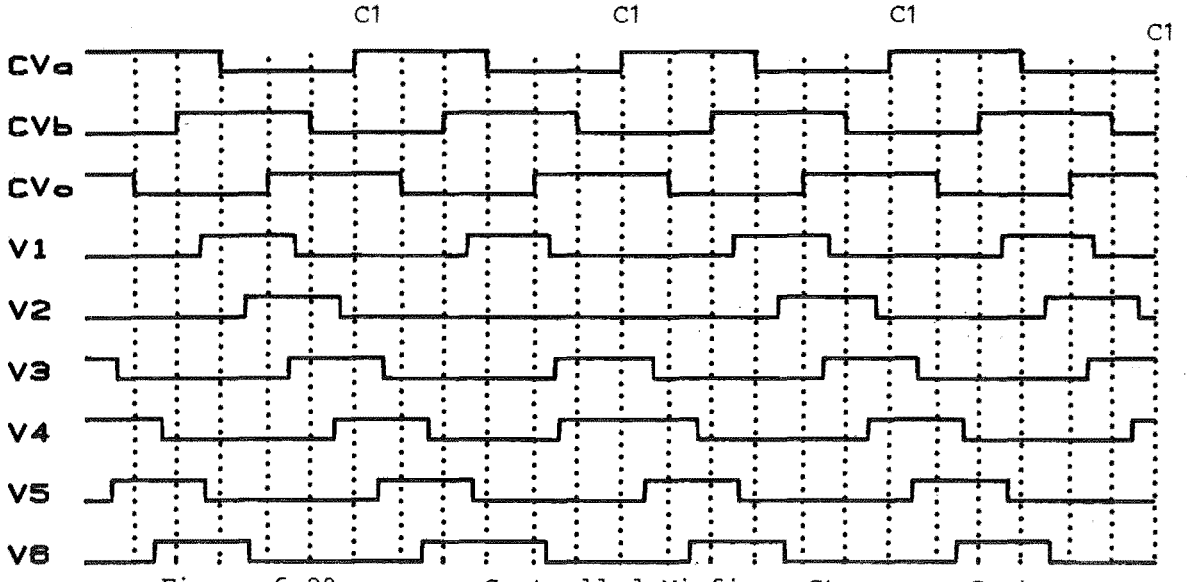
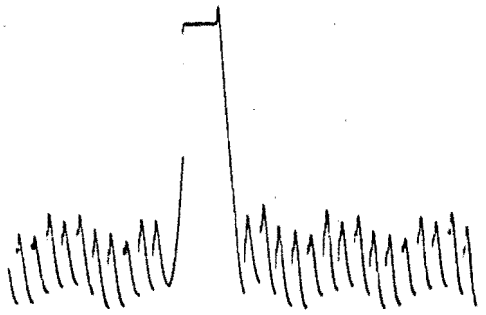


Figure 6.20 Controlled Misfire Strong ac System

In practise this technique is not particularly useful because the actual disturbance created by the fire through is relatively small and the cumulative effect of firing through Vi-1 in response to a fire through of Vi is often larger than the uncontrolled fault. This is apparent in fig 6.17, where a large disturbance in the voltage waveform is visible.

The normal conduction pattern resulting from an uncontrolled misfire was shown in fig 6.8 while fig 6.19 shows the corresponding voltage waveform.



Uncontrolled Misfire

Figure 6.19

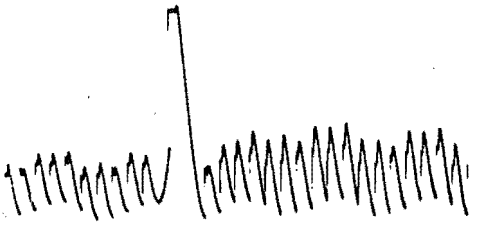


Figure 6.21 Controlled Misfire
Strong ac System

The effectiveness of fault development control techniques for misfires depends on the type of fault detection technique used. If the technique based on using the unexpected turn-on of valve Vi+1 to detect the misfire of valve Vi is used, then the sixty degree delay in detection time has serious consequences for the success of the fault development control. On a strong system it is possible to fire through valve V4 to commutate off valve V6 and the commutation should complete before the commutating voltage reverses polarity at crossing C6. If this is attempted then the conduction patterns of fig 6.20 and the waveforms of fig 6.21 result.

However on a weaker system the responses shown in figs 6.22 and 6.23 are more typical.

The fault is detected at point A and valve V4 is fired. The commutations V1 to V3 and V6 to V4 both fail despite the commutation overlap extending to about eighty degrees. The net result is that valves V1 and V6 remain on and a further consequential commutation failure in valve V2 followed by a consequential misfire in valve V4 are recorded before normal operation resumes. The major reason for the commutation failures in valves V3 and V4 appears to be that , during the fault, the commutating voltages are very distorted and the commutation overlap increases considerably. This tends to confirm the objections outlined in section 6.4.1 above.

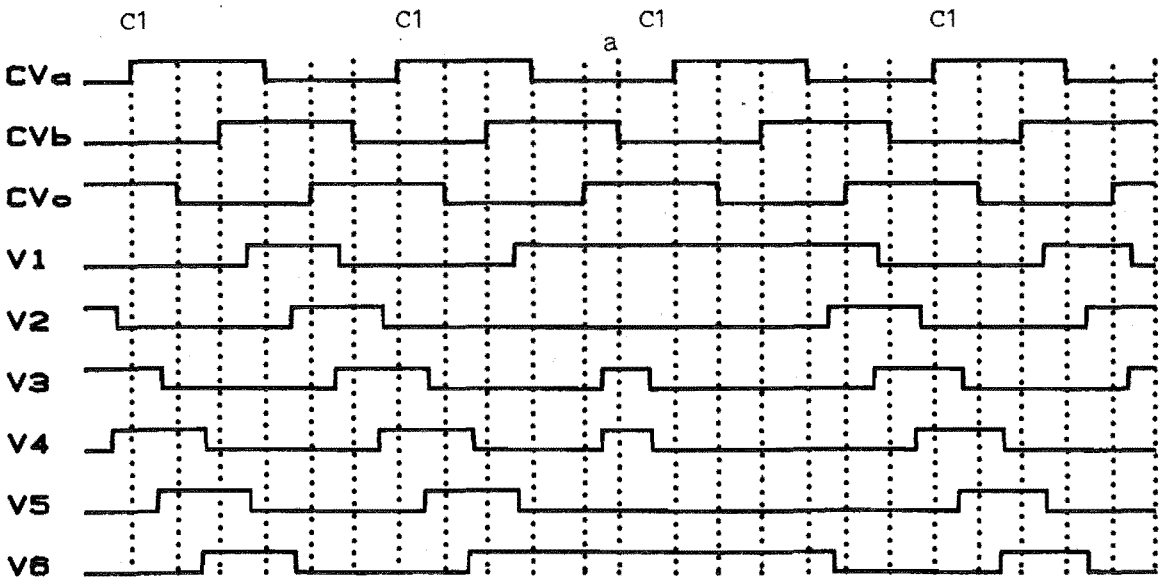


Figure 6.2 Controlled Misfire Weak ac System Control Fails

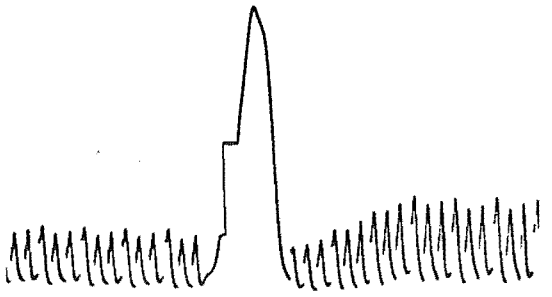


Figure 6.23 Misfire Control Failure

Controlled Misfire Alternative Detection

Figure 6.24

If however the alternative misfire detection technique is used then the detection of the misfire is made some sixty degrees earlier and there is now sufficient time for the commutations of V1 to V3 and V6 to V4 to complete. The results obtained in this case are shown in figs 6.24 and 6.25 respectively.

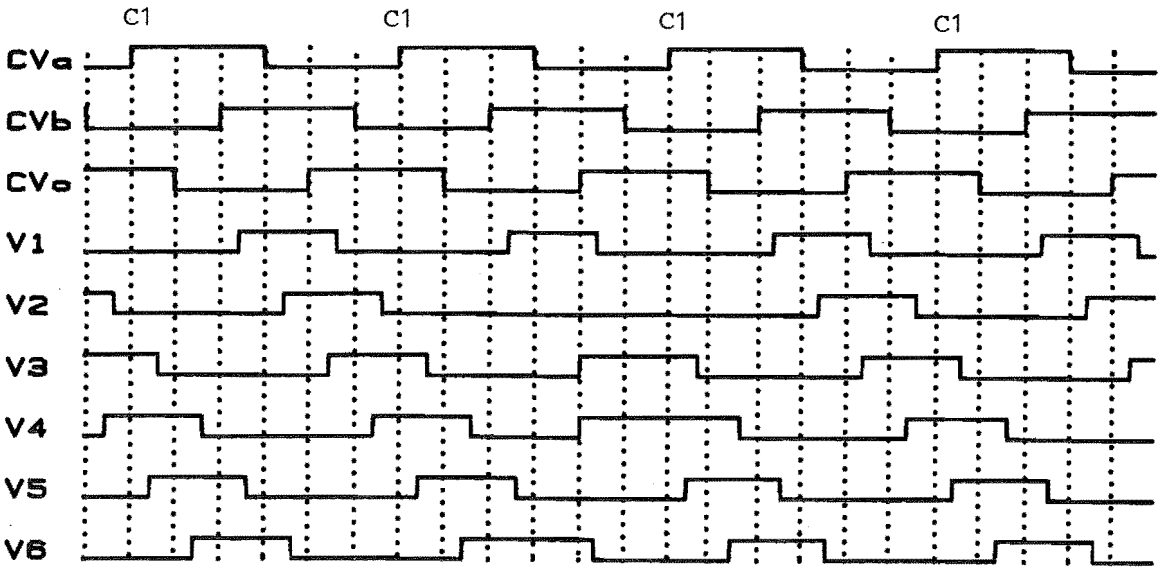
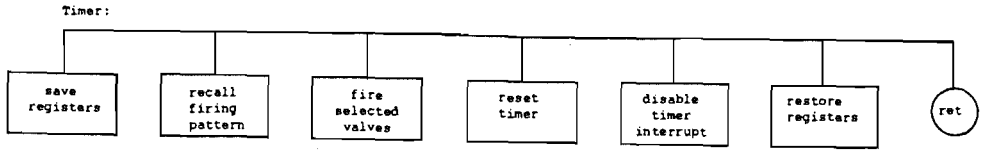


Figure 6.25 Controlled Misfire on Weak ac System
Alternative Misfire Detection

Note that both valve V4 and valve V3 must again be fired through simultaneously otherwise a short circuit between valves V1 and V4 will result. If this pair is fired through immediately then the invertor voltage will rise considerably and the transient that will result will be considerably larger than would have resulted for the uncontrolled development.

Consequently on detection of the fault the valves that must be fired through are calculated, but the valves are not fired immediately. Instead, a delay timer is started which delays the firing for approximately forty degrees. This was shown in fig 6.11, and the response to the interrupt generated on the timer reaching terminal count is shown in fig 6.26.



Alternative Misfire Detection Interrupt Routine
Figure 6.26

The selected valves are recalled and the appropriate firing commands issued. The timer is then restarted to remove the interrupt request and subsequent interrupts at this level are disabled. The bar graph of fig 6.27 and the voltage waveform of fig 6.28 show the effect of insufficient delay in the fault development control routine. Observe that valves V3 and V4 are fired considerably earlier than was shown in fig 6.24. The result of this is readily apparent when the waveform in fig 6.28 is compared with that of fig 6.26.

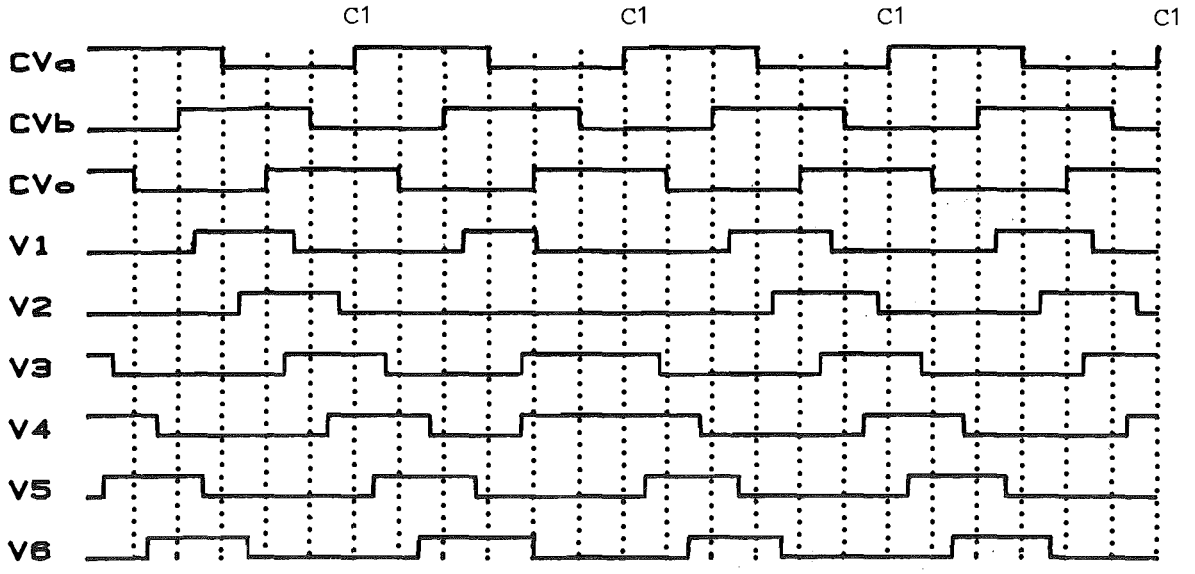
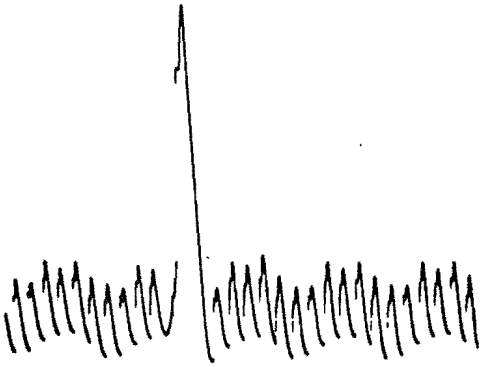


Figure 6.27 Controlled Misfire Alternative Detection Insufficient Delay

Further results examining the effectiveness of fault development control on other faults, such as double successive commutation failures, and on weaker ac systems will be presented in chapter 7.



Controlled Misfire Alternative
Detection Insufficient Delay

Figure 6.28

7.0 Factors Affecting The Operation of Fault Development Control.

7.1 Effect of Short Circuit Ratio on Fault Development Control.

As outlined in chapter 6.4.2, the ac system impedance can have a large influence on the success or failure of the proposed fault development control techniques. This section will examine how fault development control operates on systems with short circuit ratios in the range two to twenty. It will also examine the effect of harmonic filters on the operation of the fault development control system. Figure 7.1 shows a simplified line diagram of an ac/dc system terminal.

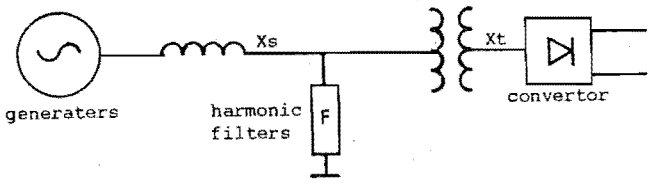


Figure 7.1 Convertor Line Diagram

The factor that primarily determines the success or failure of the fault development control techniques is the commutation time required to successfully perform the necessary valve switchings. Under steady state conditions the commutation overlap is given by

$$\mu = \cos^{-1}\left[\cos\alpha - \frac{\sqrt{2}\, Xc\, Id}{E}\right] - \alpha$$

Consider the case of an invertor, operating with the controller in constant extinction angle mode, and driven by a rectifier with a constant current controller. The dc current, Id, may be considered to be constant and any increase in overlap must produce a reduction in delay angle in the steady state. During the initial stages of the fault the invertor's controller is physically incapable of altering the delay angle to compensate for the fault and any change in commutation angle must be reflected in a reduction in extinction angle.

Under steady state operation the commutating reactance Xc is constant and, in the presence of ac filters, is basically the leakage reactance of the transformer Xt (Kimbark, 1971). If filters are not present in the ac system then the commutating reactance must include both the transformer leakage reactance and the ac system reactance Xs.

Under fault conditions the situation is similar. If the filters can maintain the sinusoidal bus voltages then the commutating reactance remains the transformer leakage reactance. If for any reason the filters are incapable of maintaining a sinusoidal bus voltage then some of the system reactance must be added to the commutating reactance. In the absence of filters, the system reactance again must be taken into account, and usually it will be necessary to go further back into the ac system to find a sinusoidal voltage under fault conditions than in the steady state. The overall effect of system reactance on the commutation overlap is shown in figure 7.2.

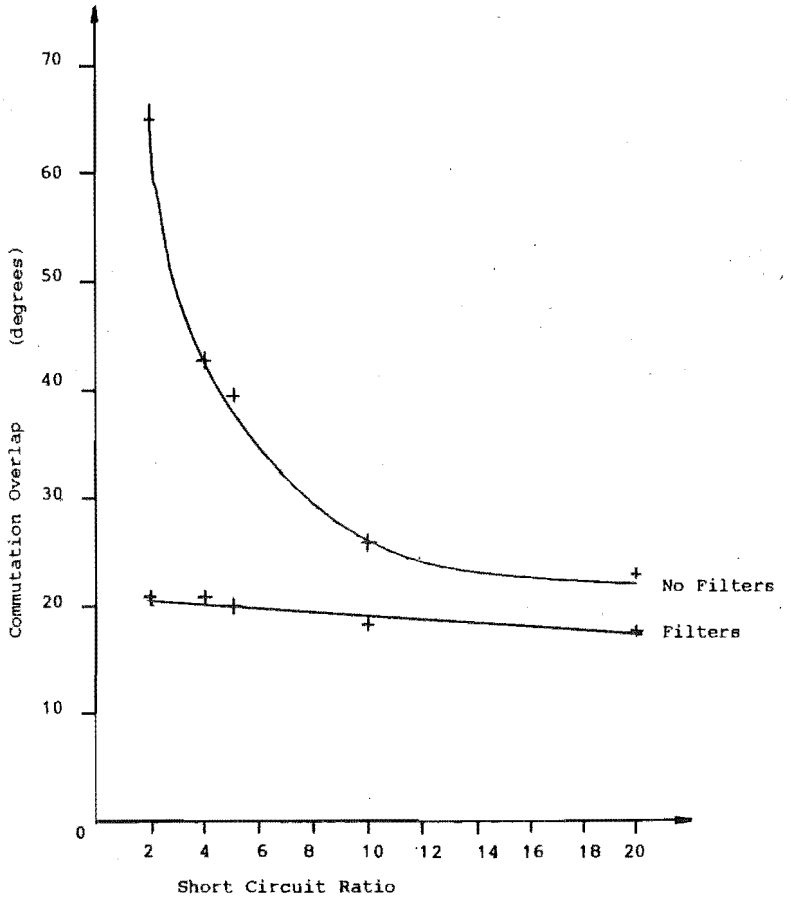


Figure 7.2 Effect of SCR on Overlap

This graph was obtained by recording the measured commutation angle for various strength systems, with the harmonic filters in and out of circuit. As is apparent , because the filters are able to maintain an undistorted bus voltage , the commutation time only increases slightly in the presence of filters, but in the absence of filters the commutation time increases considerably.

The overall effect is that any commutation will take longer on a weaker system than it would on a stronger system. As will be shown, any significant increase in commutation time will seriously degrade the operation of the fault development control techniques.

7.1.1 Fault Development Control of Commutation Failures on Weak Systems.

The uncontrolled development of a single commutation failure in valve V2 has been described in chapter 6.3.3.1. For convenience the bar graph associated with this fault is repeated in fig 7.3.

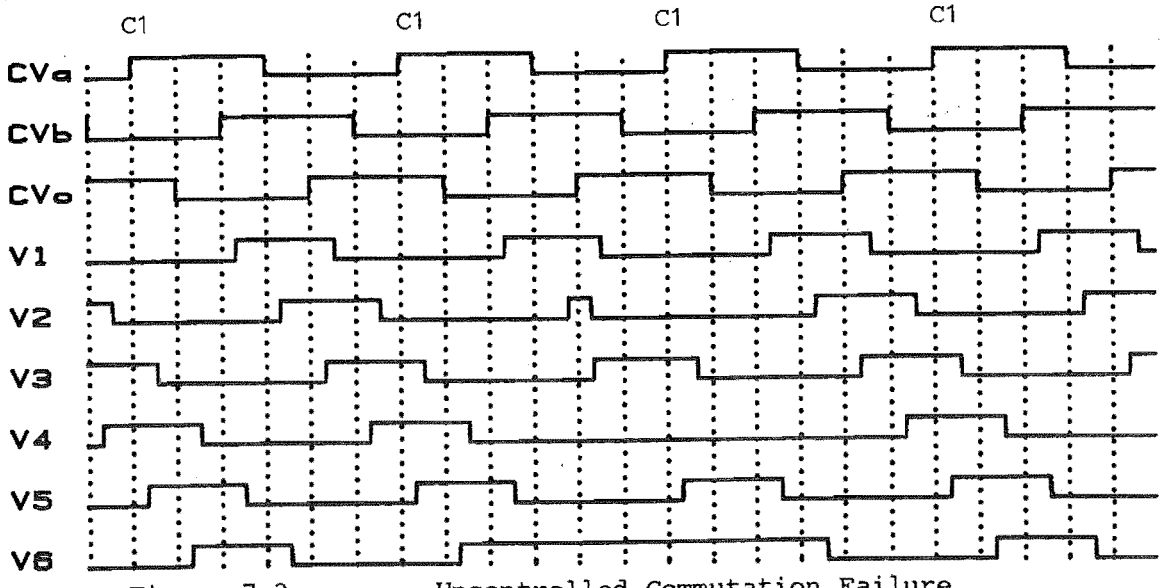
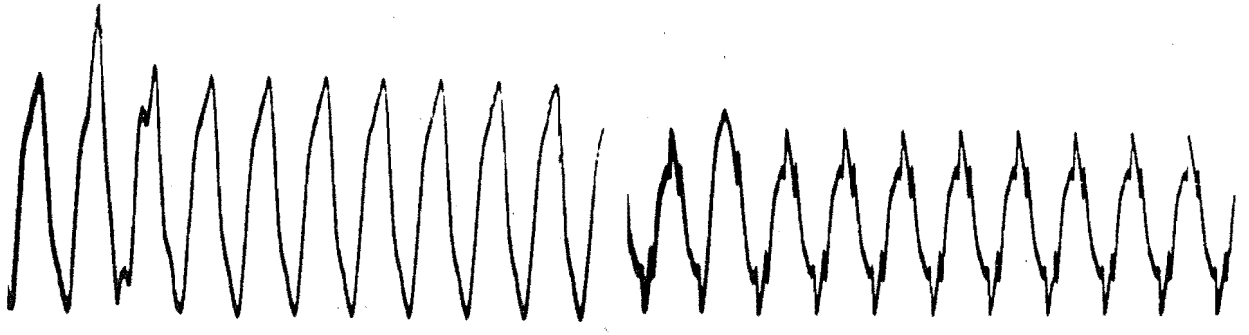


Figure 7.3 Uncontrolled Commutation Failure
Strong ac System

By examining the relationship between the indicated valve turn-offs and the commutating voltage crossings in fig 7.3, it can be seen that the recorded result was obtained from a convertor acting as an invertor, with a combined commutation and extinction angle of about thirty degrees. Of this thirty degrees, approximately ten degrees are due to the commutation overlap, while the remaining twenty degrees form the extinction angle. As a result of the commutation failure valve V6 has remained on and the fault development control technique proposed in chapter 6 would correct this by firing through valve V4 to commutate off valve V6.

As explained in chapter 6, detection of the fault will occur about five degrees after the unexpected turn off of valve V2, ie about ten degrees after the C5 crossing. If valve V4 is to successfully commutate off valve V6 then the commutation must complete before the red-blue commutating voltage reverses polarity at crossing C6. There is, therefore, a period of about fifty degrees in which this must occur. Based on the graph of fig 7.2 this should be an adequate interval in the presence of ac filters, but may prove to be insufficient on weak systems in the absence of filters.

Results obtained from the equipment show that this is not the case. As expected, in the absence of filters, the development control technique works adequately on systems down to a short circuit ratio of five. On systems with a short circuit ratio of four or less the operation becomes progressively more erratic until it ceases to work entirely on a system of a short circuit ratio of two. However, in the presence of harmonic filters the operation of the fault development control technique has proved to be erratic on all systems with short circuit ratios less than ten. This appears to be due to extended commutation times caused by severe distortion of the commutating voltage waveforms during the fault. This distortion seems to be due to a resonance between the capacitance of the filters and the inductance of the ac system impedance model. The photographs of figure 7.4 demonstrate this distortion. Both photographs were taken with a short circuit ratio of four during the simulation of an uncontrolled commutation failure.



Convertor Bus Voltage
Filters Present
Figure 7.4

Convertor Bus Voltage
No Filters

Fig 7.4a was taken with the harmonic filters in circuit, while fig 7.4b was taken with them removed. The apparently different magnitudes in the two waveforms result from the differing amounts of reactive compensation present in each case. With the filters out of circuit the distortion due to the valve commutations is very apparent. However the increase in voltage distortion due to the fault is much worse when the filters are present, and this must be put down to the resonance experienced on the weaker systems. As explained in section 7.1, if the commutating voltages become badly distorted then the commutation time is lengthened, and this leads to the erratic operation recorded.

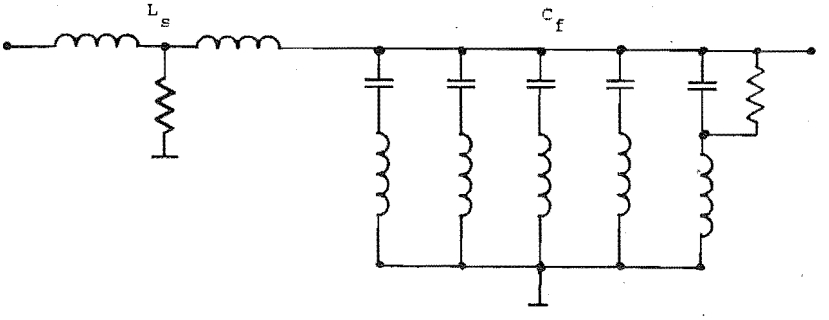


Figure 7.5 Filter Configuration

The filters used in the convertor model are typical of those used on earlier convertors, and consist of single tuned arms for the 5th, 7th, 11th and 13th harmonics, with a damped high pass network for the higher order harmonics. The combined ac system model and harmonic filters are shown in figure 7.5 . Below the resonant frequency each arm of the filter appears capacitive, and this capacitance resonates with the inductance of the ac system. The problem is accentuated in this particular model because, for reasons of limited physical space, the filters were designed to provide 100% reactive compensation instead of the more usual 40%-50%. This extra compensation makes the filters appear more capacitive than they normally would and consequently the resonance with the ac system occurs at lower frequencies than would normally occur. Theoretical calculations indicate that , with a short circuit ratio of two, a resonance near the second harmonic will occur, while with a short circuit ratio of four, the resonance will be almost exactly at the third harmonic.

This particular filter configuration has given problems in some situations and has lead to the development of the 'C' type filter (Stanley et. al., 1977), which is now starting to appear in newly commissioned schemes (Brewer et. al., 1981). The 'C' type filters consist of a single damped arm, and it appears that this configuration is less prone to the resonance problems experienced by the older style filters. Unfortunately filter modules of this 'C' type configuration are not yet available for the simulator being used and so it is not known how the fault development control would fare if these were fitted.

The indicated resonances near the second and third harmonics were confirmed by experimental results obtained from a power system harmonic analyser (Wandel and Goltermann NOWA-1). During a fault , with the filters in circuit and a short circuit ratio of two, the second harmonic voltage on the filter bus was observed to rise to a level approximately 350 times greater than that present during normal operation. Repeating the test with filters out of circuit produced a rise in second harmonic voltage of only 10 times the pre fault level. For an ac short circuit ratio of four a resonance near the third harmonic was also confirmed by experimental measurements.

Due to the problems experienced with filter - system resonance on this particular convertor model, the fault development control techniques for a commutation failure cannot be said to be a total success. Filter system resonance is not yet a major problem on most commercial schemes,although as newer schemes are commissioned on weaker ac systems it may become more of a problem. Consequently it is possible that the techniques described may still prove as useful on weak systems as they appear to be on stronger systems.

7.1.2 Fault Development Control of Misfires on Weak Systems.

The natural development of the misfire is similar to that of the commutation failure, with the important difference that the faulted valve never begins conduction rather than fails to maintain conduction. Fig 7.6 shows the natural development of a misfire in valve V2.

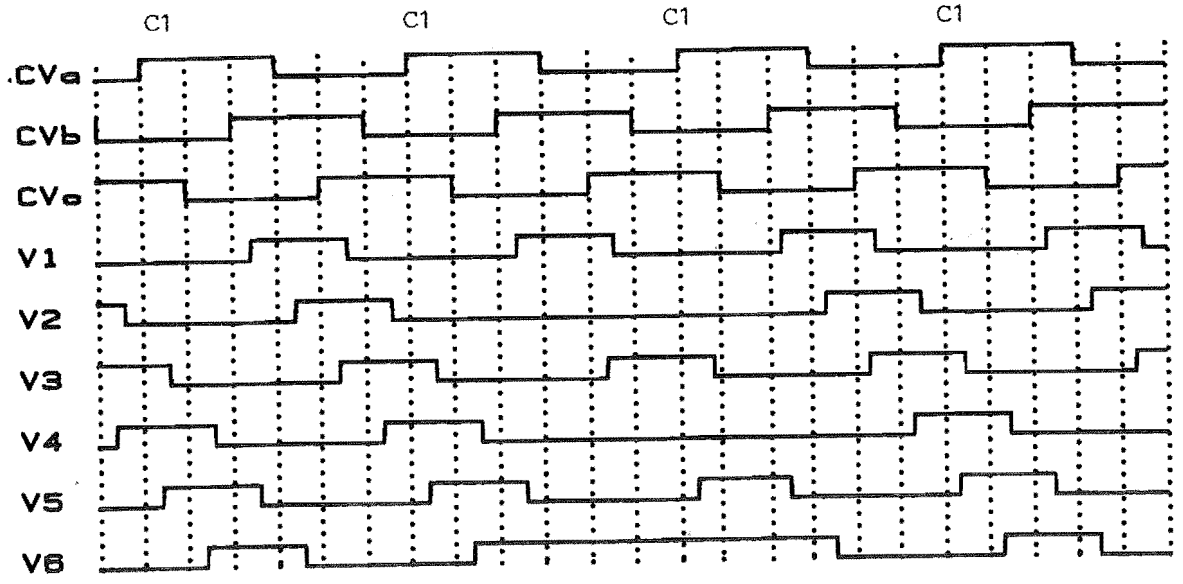


Figure 7.6 Uncontrolled Misfire Strong ac System

As in the example of the commutation failure presented above , this was recorded from an invertor operating with a commutation angle of about ten degrees and an extinction angle of about fifteen degrees. Again, as a result of the misfire, valve V6 remains on and if it is to be successfully commutated off by valve V4 then the commutation must complete by the C6 crossing. The amount of time actually available for the commutation depends on the type of fault detection procedure used.

If the misfire technique originally proposed in chapter 6.2 is used then the fault will be detected about five degrees after the unexpected turn-on of valve V3. This turn-on occurs about thirty degrees before crossing C6 and effectively leaves only about twenty five degrees for the commutation to take place and for valve V6 to regain its blocking ability. The graph of fig 7.2 indicates that this should be just adequate in the presence of harmonic filters, and this has been confirmed for strong to medium strength systems by the results obtained. On weak systems, the filter - system resonance effects described in section 7.1.1 result in extended commutation times and the development control technique fails. The technique also works adequately on strong systems in the absence of harmonic filters, although as the ac system is made weaker the point is soon reached at which the commutation from valve V6 to valve V4 fails. This is shown in the bar graph of fig 7.7.

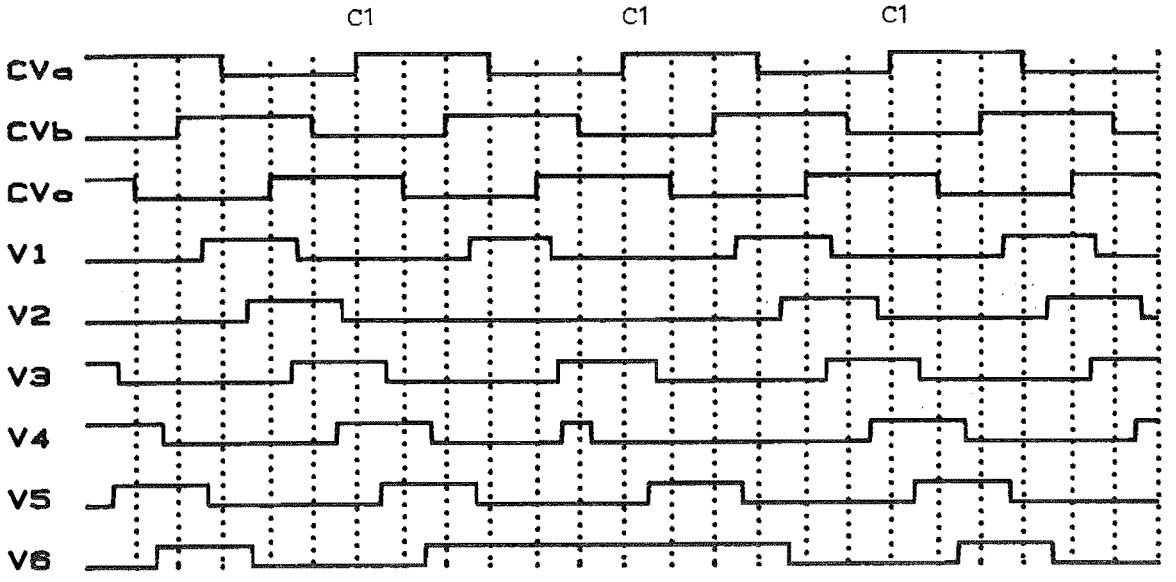


Figure 7.7 Controlled Misfire Moderate ac System
No Harmonic Filters

If the alternative misfire detection method, described in section 6.3.2 , is used then the fault will be detected about eighty degrees before the C6 crossing. This is adequate time for the fault development commutations to complete on all but the very weakest systems, with or without the presence of ac filters, and the misfire fault development control technique has proved to be extremely successful when coupled with this faster form of fault detection.

7.2 Other Typical Convertor Faults.

The results presented so far, both in this chapter, and in chapter six have all involved the simulation of single faults. This has been primarily for reasons of clarity in explaining the operation of the fault detection and fault development control sub-systems, and in explaining the results obtained from these sub-systems. In this section some other typical convertor faults will be presented, and the effect of fault development control on them examined.

7.2.1 Double Successive Commutation Failure in an Invertor.

This fault, as its name implies, involves the occurrence of commutation failures in two valves in succession.

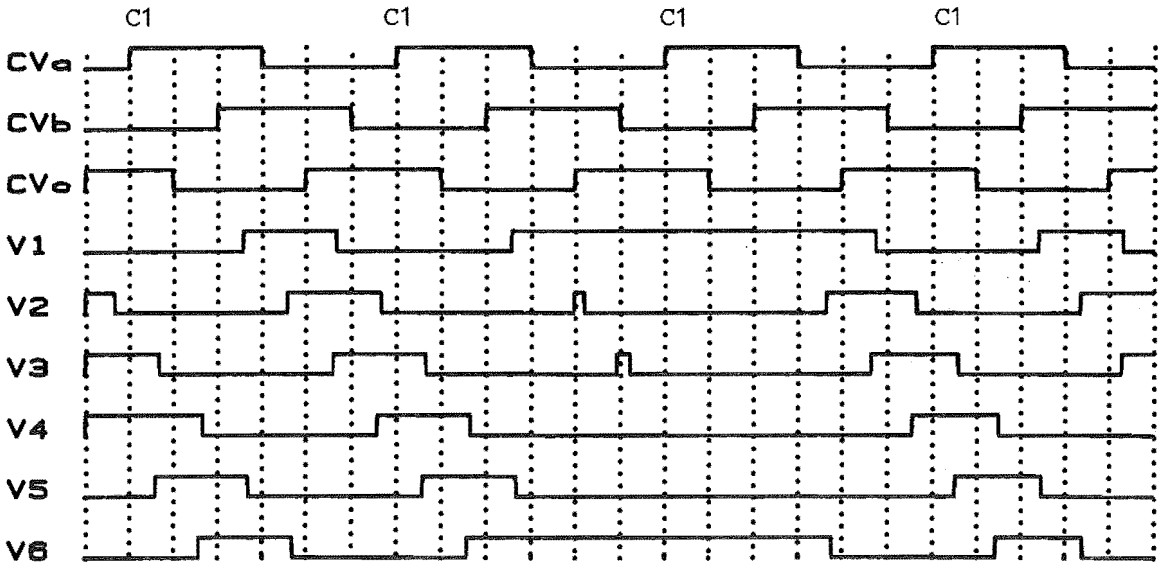


Figure 7.8 Uncontrolled Double Successive Commutation Failure Strong ac System

Figure 7.8 shows the conduction pattern of such a fault involving commutation failures from valves V6 to V2 and valves V1 to V3. Figure 7.9 shows the corresponding voltage waveform. The important feature of the uncontrolled development is that, as a result of the fault, both valves V6 and V1 remain on for an extended period of time and the terminal voltage of

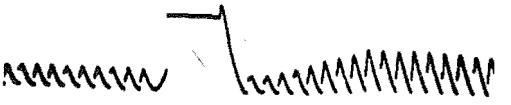
the invertor therefore follows the blue-red phase-phase voltage during this time and consequently the terminal voltage reverses polarity for a period during the fault. This is clearly visible in fig 7.9.

By using the fault development techniques described for the single commutation failure it is possible to modify this characteristic considerably. The effect of this action is presented in figures 7.10 and 7.11.

As soon as the first commutation failure in valve V2 is detected then valve V4 is fired through to commutate valve V6 off. This succeeds, although it does establish a dc short circuit as both valve V1 and V4 are now on.

Uncontrolled Double Successive Commutation Failure

Figure 7.9



Controlled Double Successive Commutation Failure

Figure 7.10

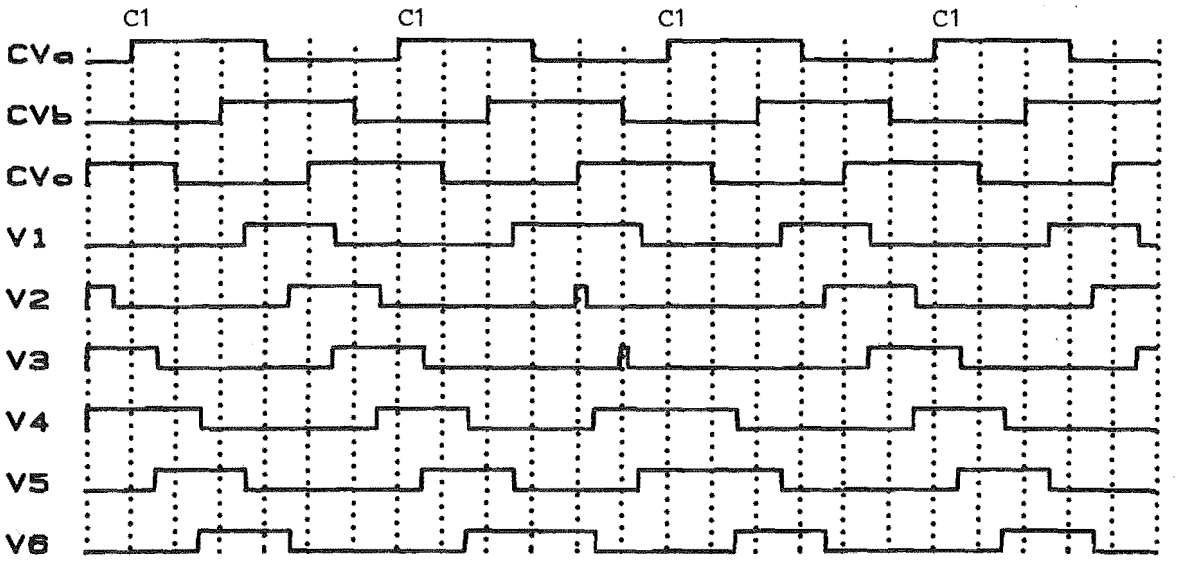


Figure 7.11 Controlled Double Successive Commutation Failure Strong ac System

Normal fault development techniques would require that both valves V4 and V3 were fired through, and in fact a firing pulse to valve V3 will have been issued by the fault development control processor. Unfortunately the method used to simulate a commutation failure in a valve actually blocks that valve until just before the reversal in the commutating voltage takes place. Due to the structure of the hardware firing circuits this block command overrides any valve fire command, and consequently the second commutation failure being simulated in valve V3 prevents it from responding to the firing pulse issued by the fault development control processor.

Following the detection of the second commutation failure in valve V3, valves V4 and V5 are fired through commutating off valve V1, removing the dc short circuit, and re-establishing normal operation. The establishment, and eventual removal of the dc short circuit is readily apparent in the voltage waveform in fig 7.10.

During a real commutation failure, where the fault occurs naturally, rather than being artificially created, the fault development control would in general be even more successful. Assuming that the second commutation failure, in valve V3 in the above example, occurs due to causes other than a fault in the actual valve firing circuitry, then the action taken by the fault development control in response to the initial commutation failure will be to fire through both valves V3 and V4. In this case valve V3 is able to start conduction and the V1 - V4 short circuit will not be established. In addition, because valve V3 has been fired through earlier than the controller would normally do, the probability of it experiencing a commutation failure is extremely small. Thus the original fault of a double successive commutation failure involving a complete reversal of convertor voltage can be controlled in to a single commutation failure with the resulting dc short circuit being limited to a very short duration. The effect of this is shown in fig 7.12.

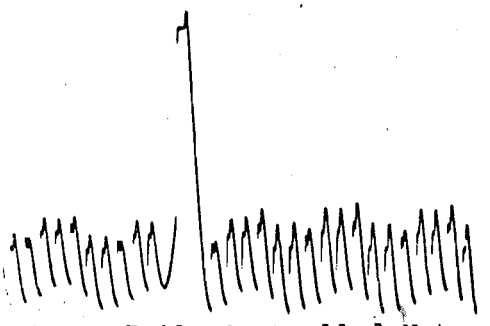
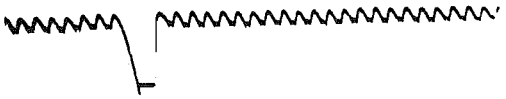


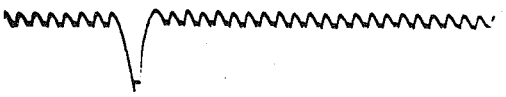
Figure 7.12 Controlled Natural Double Successive Commutation Failure

7.2.2 Misfires in a Rectifier.

So far the results presented have concentrated on faults occurring when the convertor is operated as a invertor. This has been done because such convertor faults are more likely to occur during invertor operation. The misfire is the most common type of fault that occurs in a rectifier. Commutation failures are extremely rare, except in cases of major firing circuit failure, while fire throughs, if they do occur, cause relatively little disturbance since the convertor is already being operated at low values of delay angle.



Uncontrolled Rectifier Misfire
Figure 7.13



Controlled Rectifier Misfire
Figure 7.15

The effect of a single misfire in valve V2 is presented in figures 7.13 and 7.14. These results were obtained by operating the convertor with a delay angle of eight degrees, and a commutation overlap of about seventeen degrees. As can be seen in fig 7.13, the misfire in valve V2 again leaves V6 in conduction, and when valve V3 is fired by the controller then a short circuit will be established across the dc line. This is readily apparent in fig 7.14.

The duration of the short circuit is shorter than that which occurs during the corresponding fault in an invertor. This is because the valve V6 will be commutated off by valve V4 when it is fired by the controller, approximately sixty degrees after Valve V3 was fired. This limits the short circuit duration to sixty degrees, as opposed to 120 degrees in the invertor.

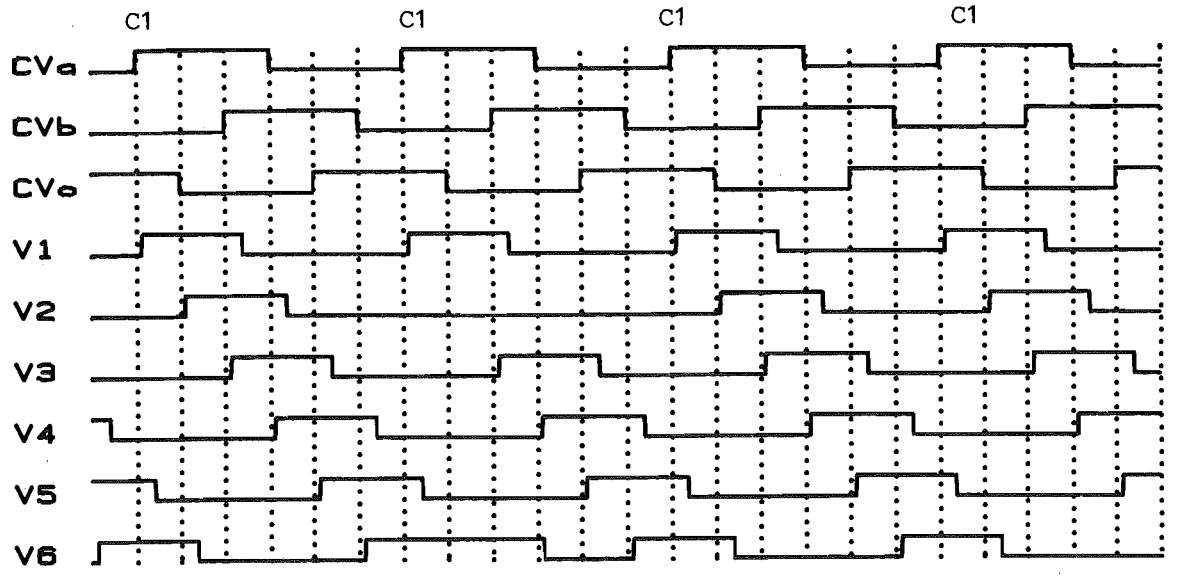


Figure 7.14 Uncontrolled Rectifier Misfire
Strong ac System

The same fault development control techniques that were proposed for use in an invertor may also be used here. The major difference is that when operating as an invertor, valve V4 may be fired through immediately because it is already forward biased. When operating as a rectifier, especially when operating at low values of delay angle, this is not always the case. In the example presented here valve V4 cannot begin conduction until the yellow-red phase to phase voltage becomes negative. This does not occur until crossing C3. The effect of this is to remove most of the advantage from the operation of the alternative misfire detection technique described in section 6.3.2.2 It is still possible to use this alternative technique, and the occurrence of the misfire will be detected about thirty degrees after crossing C2. The firing command to valve V4 can be issued at this point, but valve V4 cannot start conduction until crossing C3 for the reasons outlined above.

The original misfire detection technique will detect the occurrence of a misfire in valve V2 about five degrees after valve V3 has turned on, or about thirteen degrees after crossing C3. Valve V4 is now forward biased, and may be fired through directly upon detection of the fault. The difference between the two techniques is therefore relatively minor. The effects of fault development control on a misfire in a rectifier are shown in figures 7.15 and 7.16.

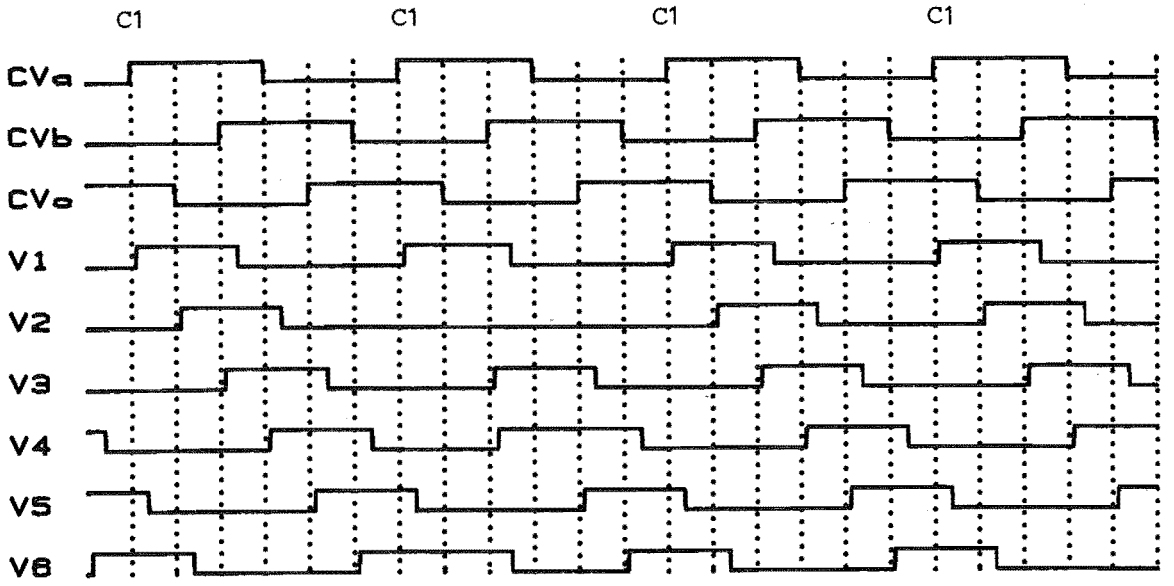


Figure 7.16 Controlled Rectifier Misfire
Strong ac System

As can be seen in figure 7.15, the duration of the short circuit has been reduced, although the long commutation overlaps that occur at low delay angles result in the reduction in short circuit duration not being as short as it could be.

7.2.3 Response to Single Phase AC faults.

Single and Multiple phase to phase and phase to neutral faults may be simulated by means of the techniques described in chapter 6.2. This section presents the results of a simulation of a blue phase to neutral short circuit. The fault is simulated by closing a relay for one cycle, and then allowing it to open naturally. The overall duration of the fault is therefore dependent on the time taken for the relay to open. The first set of results, presented in section 7.2.3.1, shows the natural response to

the short circuit for an invertor operating into a strong system, with a short circuit ratio of 20, and how the application of fault development control can alter the response. The second set of results, presented in section 7.2.3.2, shows the natural response to the fault for an invertor working into a much weaker system, with a short circuit ratio of 4.

7.2.3.1 Response to an AC Fault with a Strong AC System.

The bar graph of figure 7.17 shows the natural response of the invertor to a single phase to neutral short circuit.

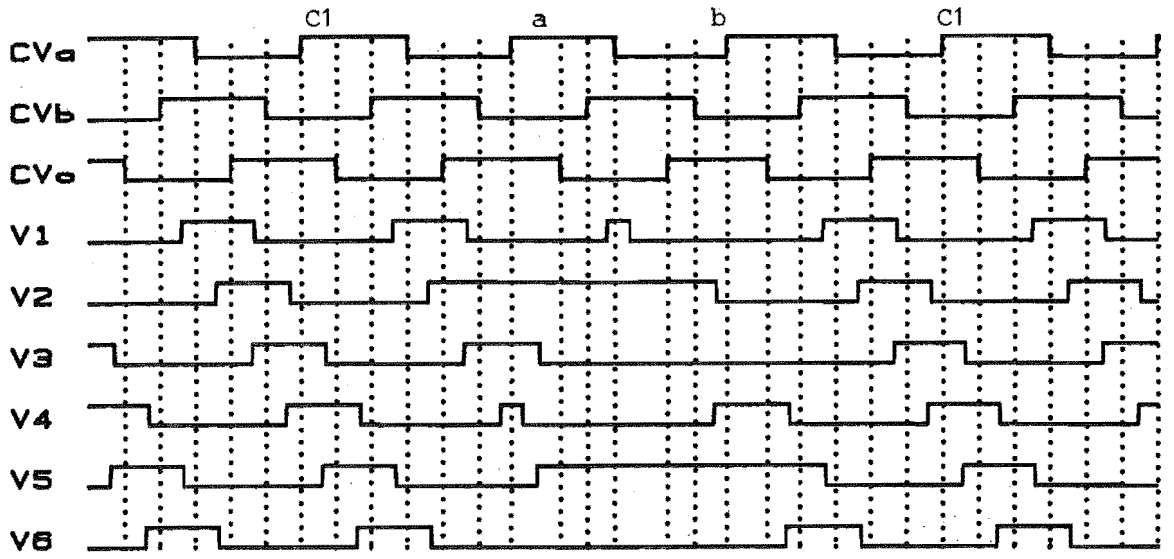
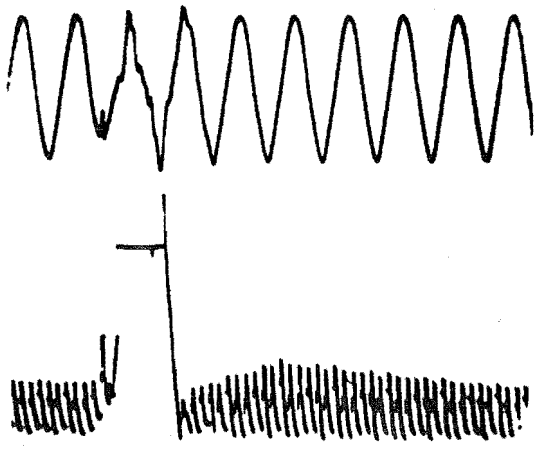


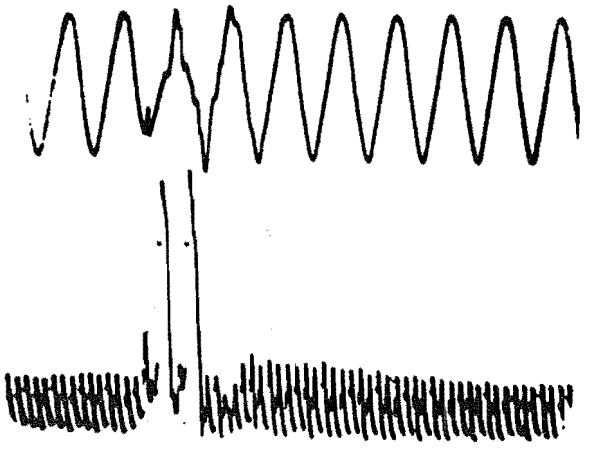
Figure 7.17 Single Phase ac Fault Strong System
Fault Development Control Inactive

Figure 7.18 shows the corresponding dc waveform on the lower trace, and the faulted ac phase voltage on the upper trace. It is readily apparent from this upper trace that the blue phase voltage becomes badly distorted during the fault. This affects the zero crossings of both the blue-red and yellow-blue commutating voltages. The effect of the collapse is graphically illustrated in the bar graph of figure 7.17. Note how, near point A, the blue-red crossing C1 moves closer to the previous C6 crossing, while the yellow-blue C2 crossing moves closer to C3. Similarly the next C4 crossing also moves closer to the C3 crossing and the C5 crossing moves closer to C6.

The displacement of crossing C1 reduces the extinction angle of the outgoing valve V2. The result is that it has not turned off before the crossing and consequently becomes re-forward biased and resumes conduction. Valve V4, which should have replace V2 turns back off. As a result of this commutation failure, valve V6 experiences a consequential misfire.



Uncontrolled Unbalanced AC Fault
Strong System
Figure 7.18



Controlled Unbalanced AC Fault
Strong System
Figure 7.19

A similar sequence of events is experienced as a result of the displacement in the C4 crossing. This time valve V1 fails to commutate valve V5 off in time and the consequential misfire occurs in valve V3.

The displacement in the C2 and C5 crossings do not cause any faults because they increase the extinction angle of the outgoing valve rather than decrease it.

Normal operation resumes when the ac short circuit is removed and the blue phase voltage amplitude returns to normal. The crossings return to their nominal spacings and when valve V4 is next fired by the controller at point B the commutation is able to complete and valve V2 is finally commutated off. During the fault both valves V2 and V5 remain on for extended periods of about 480 degrees, and a short circuit across the dc line exists for about 300 degrees.

The photograph of figure 7.19 shows the effect of the application of fault development control to an identical fault, while the equivalent bar graph is presented in figure 7.20. As in figure 7.17, the displacement of the zero crossings is evident. Again a commutation failure in valve V4 occurs, but this is successfully detected and valves V5 and V6 are fired through to compensate. This is reflected in the lower trace of figure 7.19 where it can be seen that the dc short circuit, which is established while

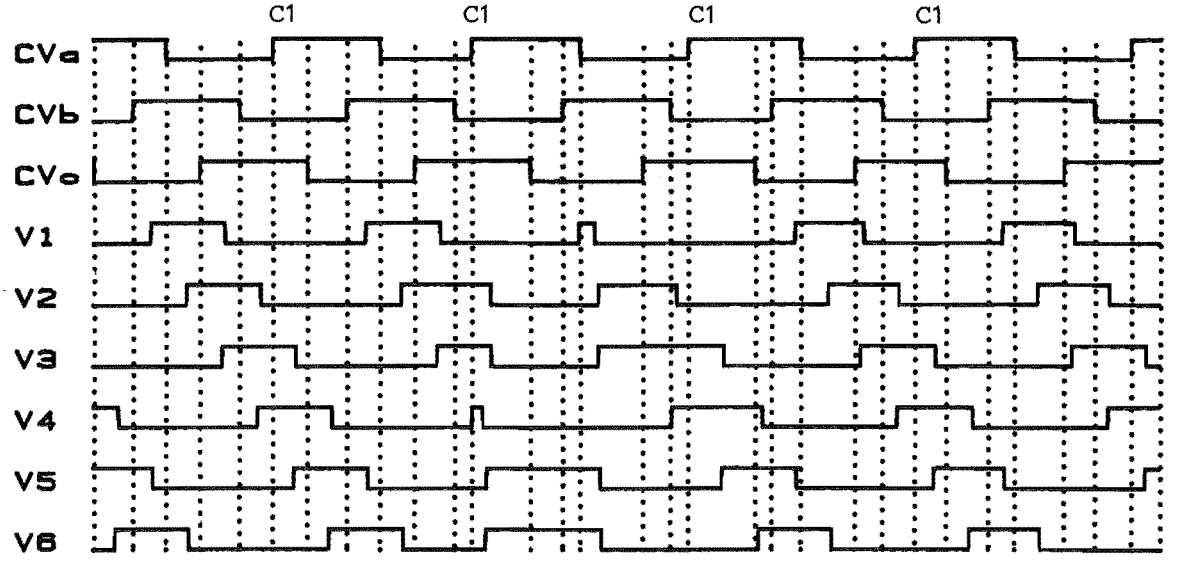


Figure 7.20 Single Phase ac Fault Strong System
Fault Development Control Active

the fault development commutation takes place, is removed. The ac fault is still present however and so the second commutation failure will still occur. This is also successfully detected and valves V2 and V3 are fired to correct this. Normal operation can be said to have resumed at this point and, as the ac short circuit has been removed by now, there are no further faults.

Due to the strong system used in this example the uncontrolled development recovers naturally, and the fault development control does not make any significant improvement to this recovery. It does however reduce the disturbance transmitted to the other end of the dc link, as the duration of the short circuit across the dc line is minimised by the application of fault development control.

7.2.3.2 Response to an AC Fault with a Weak AC System.

As described above in section 7.1.1, The convertor model used has a system-filter resonance at the third harmonic when operated at a short circuit ratio of 4. This, as was also explained above, prevents the fault development control from achieving optimum results. Despite this, as will be shown, fault development control can still have a beneficial effect on the recovery from an ac fault as well as reducing the effect of the fault on the dc system.

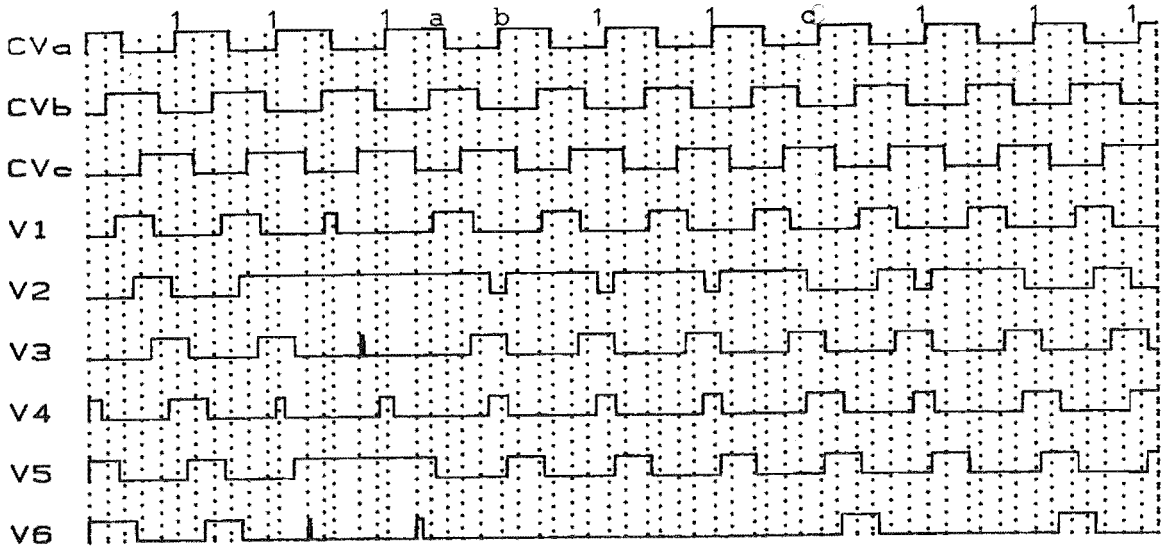
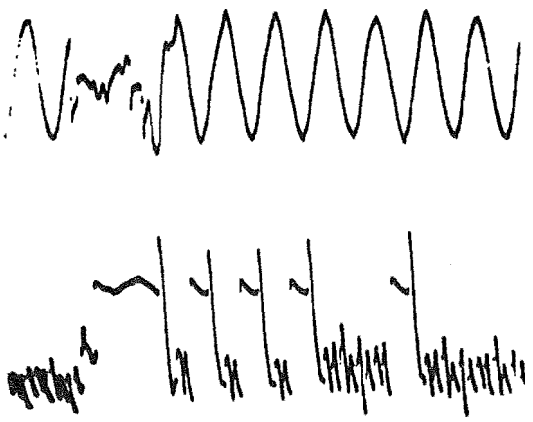
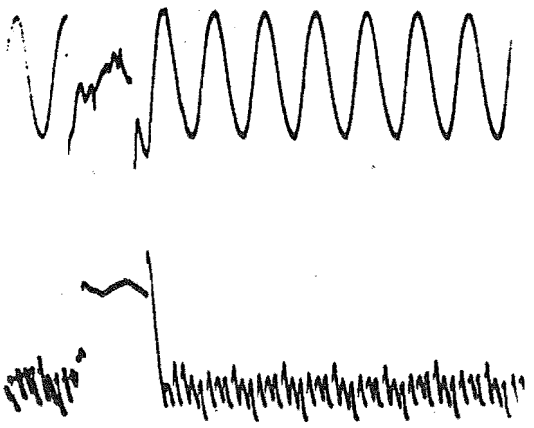


Figure 7.21 Uncontrolled Single Phase ac Fault
Weak ac System

The bar graph of figure 7.21, and the photographs of figure 7.22 show the natural response to the short circuit with a weak ac system. An immediate difference in the distortion of the ac voltage waveform is apparent when the upper traces of figures 7.22 and 7.18 are compared. This increased distortion is due in part to the weaker system, and in part to the third harmonic resonance problem. The third harmonic resonance is exacerbated in this situation by two possible causes. The first is that the imbalance condition during the fault results in the generation of even harmonics on the dc side, and the natural six pulse switching of the converter in the presence of a second harmonic current will cause the convertor to produce a large positive or negative sequence third harmonic current (Giesner and Arrillaga, 1972). A second possible source of third harmonic current will be produced when the primary of the blue phase convertor transformer, de-energised during the fault due to the collapse in the blue phase voltage, is re-energised after the fault clears. It is possible, that during the re-energisation, the transformer will go into saturation. If this occurs then a considerable third harmonic component will be present in the magnetising current of the transformer (Abu-nasser, 1974). Regardless of the source, the third harmonic current being drawn by the convertor excites the resonant circuit at its resonant frequency, and the distortion present in the upper trace of figure 7.22 results.



Uncontrolled Unbalanced Fault
Weak AC System
Figure 7.22



Controlled Unbalanced Fault
Weak AC System
Figure 7.24

The natural development of the fault starts out in the same way as did the earlier example. The displacement in the crossings due to the ac short circuit cause a total of three commutation failures, each with a corresponding consequential misfire. However when the ac short circuit is removed the recovery does not proceed in the same way in which it did in the earlier example.

Initially the displacement in crossing C4 is reduced and valve V1 successfully commutates off valve V5 at point A. This removes the dc system short circuit, and this is readily apparent in the lower trace of figure 7.22. Valve V2 remains in conduction and valve V3 switches successfully, but because of the severe distortion in the blue phase voltage, due to the resonance problem described above, the crossing C1 is still displaced. When valve V4 is fired, at point B, it initially commutates valve V2 off. However the outgoing valve does not recover sufficient blocking capacity before the C1 crossing occurs and consequently it fires through immediately after the C1 crossing. Then, when valve V5 is fired by the controller, the dc short circuit is re-established for a further 120 degrees, until valve V5 is commutated off by valve V1. This is clearly visible in the lower trace of figure 7.22. The unsuccessful V2 to V4 commutation is repeated in the following two cycles. During the next cycle, at point C, the V2-V4 commutation succeeds, only to fail again for the last time in the following cycle. As is apparent in the upper trace of figure 7.22, the distortion present in the blue phase voltage has largely died away by this time, and the convertor finally returns to normal invertor operation. From the initial commutation failure caused by the ac fault, until the eventual recovery of the convertor, a total of eight cycles have elapsed. Based on conventional fault detection and protection strategies, a fault of this magnitude would probably require the invertor to be bypassed and restarted when the disturbance had cleared.

The effect of fault development control on such a fault is shown in figures 7.23 and 7.24. As can be seen in the bar graph of figure 7.23, the ac short circuit again causes a displacement in the commutating voltage zero crossings. As a result a commutation failure of valve V4 occurs. This is detected and, as shown in fig 7.23, valves V5 and V6 are fired through to compensate for this. Both commutations successfully complete, and the dc short circuit, shown in the lower trace of figure 7.23, is

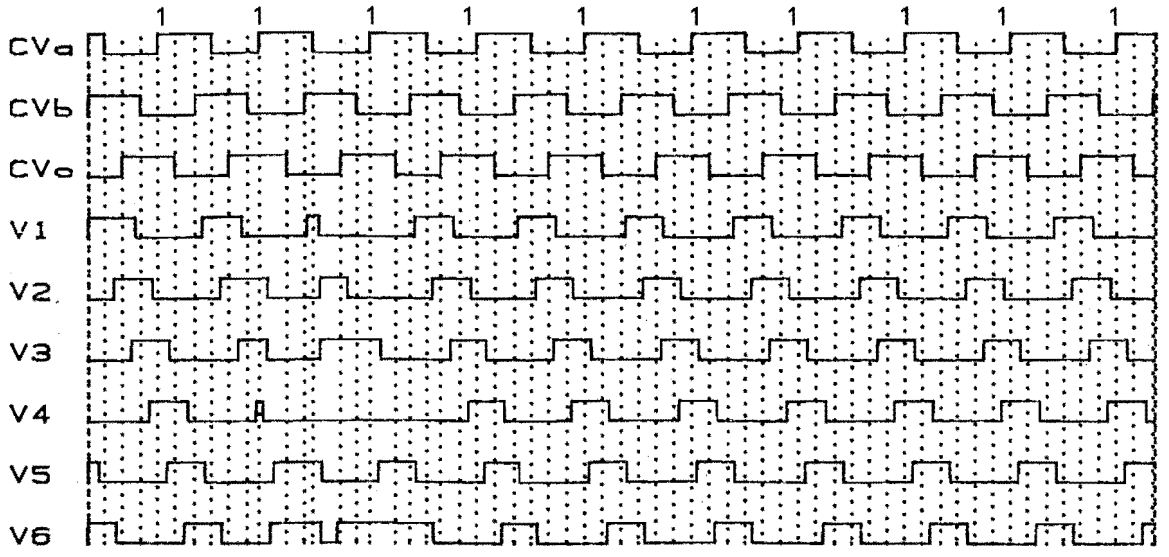


Figure 7.23 Controlled Single Phase ac Fault
Weak ac System

temporarily removed. A second commutation failure in valve V1 then occurs, and this is also detected and valve V2 and V3 fired through to correct it. Unfortunately, due to the collapse in the blue phase voltage, the commutation from valve V6 to valve V2 does not succeed, and valves V3 and V6 are left on. This re-establishes the dc short circuit until valve V5 can commutate off valve V3. Finally because of the severe distortion in the commutating voltages, valve V4 misfires. This is also detected, and valves V5 and V6 are fired through to correct it. Valve V6 is still in conduction so the repeated firing has no effect, but by firing valve V5, the dc short circuit is removed and the convertor recovers to normal operation with no further consequential faults.

A comparison of the lower traces of figures 7.22 and 7.24 shows the reduction in consequential faults due to the action of fault development control. By the application of fault development control the duration of the fault has been reduced to little more than a single cycle. This removes the necessity to bypass the convertor, and should increase the reliability of the convertor during operation.

8.0 Conclusion.

A microprocessor system for use in investigating many aspects of HVDC convertor control and protection has been developed. The system provides a powerful interactive development tool which may be used to study almost any feature of convertor operation. A comprehensive monitoring system has been included as part of the HVDC development system to allow the user to interactively assess and optimise the performance of individual components of the HVDC system.

The development of a relatively simple control algorithm has been presented, and a microprocessor based implementation described. The particular method of implementation allows the control algorithm to be easily modified. This permits the future investigation of alternative algorithms, based on either existing analogue techniques, or on modern digital control theory. While the control algorithm does not extend the analogue control techniques currently in use, it does have a greater flexibility and the use of a microprocessor as the control element would permit the development of 'intelligent' adaptive control techniques.

An improved convertor fault detection algorithm has been described. This new technique provides reduced detection times, and improved fault descrimination when compared with the more traditional approaches. A method has been presented that allows this technique to handle any sequence of convertor faults and unambiguously track the development of the overall fault sequence. The implementation of these techniques on a microprocessor has also been described, and its performance assessed.

An application of this improved fault detection technique to fault development control has been demonstrated. The basic principles of fault development control have been described and a microprocessor based implementation presented. The operation of the fault development control system has been extensively investigated, both on typical convertor fault sequences simulated in the convertor, and on fault sequences resulting from ac disturbances. The effect of increasing system impedance on the operation of fault development control has also been examined.

In general, excluding the problems associated with filter-system resonance, the fault development control techniques developed in this thesis have proved to be reasonably successful. Their success is such that I believe they may well of practical use.

If a fault detection and development control scheme were to be implemented as part of an overall microprocessor based control system then it would be possible to detect and control the initial fault in the majority of instances. As was shown in chapter 7, controlling the initial fault substantially reduces the disturbance due to that fault, and consequently increases the probability of a natural recovery. Therefore such techniques should reduce the number of forced outages and consequently improve the overall reliability of the hvdc transmission system by reducing the occasions on which a convertor must be bypassed in order to recover from a series of faults.

There exist several areas in which improvements may be made in the system described in this thesis. Of these, the most important is the general area of convertor control. Considerable scope exists for the investigation of alternate control algorithms to improve both the steady state and transient performance of the existing controller. An investigation of modern digital control techniques, and their applicability to HVDC control would also be useful. A more detailed investigation of the operation of fault development control is also needed, particularly its operation on weak ac systems. It should be possible to expand its operation, and link it with the steady state controller to form some form of 'intelligent' fault development control system.

There are also items in the hardware of the HVDC Development System that need further work. In particular, the zero crossing detector circuits, and the data acquisition methods in general can be improved. The production of some 'C type' filter modules would also aid the investigation of the operation of fault development control on weak ac systems.

Finally there is the possibility that the information recorded from the convertor might be used to detect faults occurring in the connected ac and dc systems. No work has been attempted in this area as yet, and it is possible that useful results might be achieved.

9.0 References

The references have been divided into three categories depending on the source. That is papers, books and technical manuals.

9.1 Papers.

ABU-NASSER, A.A.A.I. : "Transformer Inrush in Traction and HVDC Schemes.", UMIST, Manchester, England, 1981, (Thesis, Ph.D.: Engineering).

AINSWORTH, J.D. : "The Phase Locked Oscillator - A New Control System for Controlled Static Convertors." IEEE Transactions on Power Apparatus and Systems, Vol. PAS-87, No. 3, pp 859-865, March 1968.

AINSWORTH, J.D. : "Harmonic Instability Between Controlled Static Convertors and A.C. Networks.", Proc IEE, Vol. 114, No. 7, pp949-957, July 1968

AINSWORTH, J.D. and Bowles, J.P. : "An HVDC Transmission Simulator Suitable for Both Steady-State and Transient Studies.", Proc IEE International Conference on High Voltage D.C. Transmission, UMIST. IEE Conference Publication No. 22, pp 129-131, September 1966.

ARRILLAGA, J. and BALDWIN, D.G. : "Direct Digital Closed Loop Control of HVDC Convertors.", Proc IEE., Vol. 121, No. 12, pp 1567-1571, December 1974.

ARRILLAGA, J. DEWE, M.B. HISHA, H. and LAKE, C.B. : "Potential Applications of Microprocessors in the Control of HVDC Links.", Proc IFAC Symposium on Computer Applications in Large Scale Power Systems, New Delhi, pp 303-308, August 1979.

ARRILLAGA, J. and ERINMEZ, A. : "Direct Digital Control of Power in HVDC Interconnections.", Proc. IEE., Vol. 118, No. 12, pp 1800-1806, December 1971.

ARRILLAGA, J. and GALANOS, G. : "Fault Development Control in A.C.-D.C. Convertors.", PROC IEE, Vol. 116, No. 7, pp 1201-1208, July 1969.

ARRILLAGA, J. and GALANOS, G. : "Fault Detection Scheme for a Direct Digital Control in A.C.-D.C. Interconnections.", PROC IEE, Vol. 117, No. 4, pp 1947-1954, April 1970a.

ARRILLAGA, J. and GALANOS, G. : "Theoretical Basis of a Digital Method of Grid Control for HVDC Convertors.", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-89, No. 8, pp 2049-2055, November/December 1970b.

ARRILLAGA, J., GALANOS, G. and POWNER, E.T. : "Direct Digital Control of HVDC Convertors." ibid., Vol. PAS-89 No. 8, pp 2056-2065, November/December 1970.

ARRILLAGA, J. and GIESNER, D.B. : "Recovery of HVDC Interconnections From Disturbances in the Thyristor Bridges.", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-91, No. 6, pp 2333-2341, November/December 1972.

ARRILLAGA, J. and HISHA, H. : "Fast On/off Detection of Silicon Controlled Rectifiers and its Influence on Convertor Controllability", IEEE Transactions on Industrial Electronics and Control Istrumentation, Vol. IECI-26, No. 1, pp 22-26, February 1979.

BALDWIN, D.G. : "Digital Closed-Loop Control of HVDC Convertors.", UMIST, Manchester, England, 1972, (Thesis: Ph.D. Engineering)

BREWER, G.L., WELCH, I.M., and ABRAMOVICH, B.J. : "Harmonic Filters for the Cross Channel Link.", Proc. IEE International Conference on Thyristor and Variable Static Equipment for ac and dc Transmission, London, Conference Publication No. 205, pp 121-124, November/December, 1981.

BOWLES, J.P. : "AC System and Transformer Representation for HVDC Transmission Studies.", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-89, No. 7, pp1603-1609, September/October 1970.

DEWAN, S.B. DUNFORD. W.G. : "A Microprocessor-Based Controller for a Three-Phase Controlled Rectifier Bridge.", IEEE Transactions on Industry Applications, Vol. IA-19, No. 1, pp113-119, January/February 1983.

ELSTROM, A. and LISS, G. : "A Refined HVDC Control System." IEEE Transactions on Power Apparatus and Systems, Vol. PAS-89, No. 5, pp 723-732, May/June 1970.

GALANOS, G., VOVOS, N. and GIANNAKOPOULOS, G. : "Combined Control and Protection System for Improved Performance of HVDC Links in the Presence of AC Faults.", Proc IEE Vol. 131, Pt. C, No. 4, July 1984.

GODDARD, S.C. et al. : "The New 2000MW Interconnection Between France and the U.K. ", Paper 14-09, CIGRE, 28th International Conference, Paris, 1980.

HINGORANI, N.G. and CHADWICK, P. : "A New Constant Extinction Angle Control for AC/DC/AC Static Convertors.", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-87, No. 3, pp866-872, March 1968.

HISHA H. : "Scaled Down Modelling and Microprocessor Facilities for the Simulation and Assesment of HVDC Convertor Disturbances.", University of Canterbury, Christchurch, N.Z., 1983, (Thesis: Ph.D. Engineering).

HINGORANI, N.G. and BURBERY, M.F. : "Simulation of AC System Impeadance in HVDC Syatem Studies.", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-89, No. 5, pp820-828, May/June 1970.

JOTTEN, R. et al. : "Control in HVDC Systems - The State of the Art, Part 1, Two Terminal Systems.", Paper 14-10 presented in the name of Study Committee 14 (DC Links), CIGRE 27th International Conference, Paris, 1978.

MACHIDA, T. et al. : "Control and Protective System of HVDC Transmission by HVDC Simulator (Thyristor Convertor).", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-90, No. 6, pp2778-2785, November/December 1971.

MACHIDA, T. and YOSHIDA, Y. : "A Method to Detect the Deionisation Margin Angle and to Prevent The Commutation Failure of an Invertor for DC Transmission.", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-86, No. 3, pp259-267, March 1967.

REEVE, J. : "Logic Behaviour of HVDC Convertors During Normal and Abnormal Conditions." Proc IEE, Vol. 114, No. 12, pp 1937-1946, December 1967a.

REEVE, J. : "Direct Digital Protection of HVDC Convertors." Proc IEE, Vol. 114, No. 12, pp 1947-1954, December 1967b.

REEVE, J. and GIESBRECHT, W.J. : "Evaluation of a Microcomputer for HVDC Concertor Control.", IEEE Power Engineering Society Summer Meeting, Los Angeles, July 16-21, 1978, Paper A78 550-6.

REEVE, J. and SEVCENCO, J.A. : "An Automatic Control Scheme for HVDC Transmission Using Digital Techniques, Part I: Principles of Operation.", IEEE Transactions on Power Apparatus and Systems, Vol. PAS-91, No. 6, pp 2319-2324, November/December 1972.

REEVE, J. SEVCENCO, J.A. and VALI, E. : "An Automatic Control Scheme for HVDC Transmission Using Digital Techniques, Part II: Circuit Considerations and Performance.", ibid.,Vol. PAS-91, No. 6, pp 2325-2332, November/December 1972.

RIMMER, D.H. : " Design and Performance of Harmonic Filters for HVDC Convertors.", UMIST, Manchester, England, 1975, (Thesis: M.Sc. Engineering).

RUMPF, E and JARRETT, G.S.N. : "A Survey of the Performance of HVDC Systems Throughout the World During 1975-1978.", Paper 14-08 presented in the name of Study Committee 14 (DC Links), CIGRE 28th International Conference, Paris, 1980.

STANLEY, G.H. et al. : "Design and Performance of AC Filters for Twelve Pulse HVDC Schemes.", Proc IEE International Conference on Power Electronics - Power Semiconductor and Their Applications." London, IEE Conference Publication No. 154, pp 162-165, September 1977.

SHORE, N.L. and FRERIS, L.L. : "Minicomputer Online Control of DC Link Convertors.", Proc. IEE., Vol. 125, No. 3, pp 215-220, March 1978.

TSO, S.K. HO, P.T. : "Dedicated Microprocessor Scheme for Thyristor Phase Control of Multiphase Convertors.", Proc IEE Pt B., Vol 128, No. 2, pp 101-108, March 1981.

9.2 Books.

ARRILLAGA, J. : "High Voltage Direct Current Transmission.", Peter Peregrinus on behalf Intitution of Electrical Engineers, London, 1983, IEE Power Engineering Series No.6 .

KIMBARK, E.W. : "Direct Current Transmission, Vol 1." Wiley-Interscience, New York, 1971 .

UHLMANN, E. : "Power Transmission by Direct Current.", Springer-Verlag, Berlin, 1975 .

9.3 Manuals.

" Data Acquisition Products Catalog Supplement", pp 215-222, 1979, Analog Devices, Route One Industrial Park, P.O. Box 280, Norwood, Massachusetts 02062.

" MBC-86/12 CPU Board.", Man No. 168-A80-01/05, Matrox Electronic Systems, 5800 Andover Ave, T.M.R., Que., H47 1H4, Canada.

" SA 800/801 Diskette Storage Drive OEM Manual.", 1977, Shugart Associates, 415 Oakmead Parkway, Sunnyvale, California 94086.

The following manuals are all available from Intel Corp.,
3065 Bowers Ave., Santa Clara, Cailfornia 95051.

a " iSBC 86/12a Single Board Computor Hardware Reference Manual. ",
Man. No. 9803074-02.

b " iSBC 056a RAM Board Hardware Reference Manual. ", Man. No. 143572-001.

c " iSBC 208 Flexible Disk Controller Hardware Reference Manual. ",
Man. No. 143078-001.

d " Intellec Series II Microcomputer Development System Hardware Reference
Manual. ", Man. No. 9800556-02.

e " iSBC 80/30 Single Board Computor Hardware Reference Manual.",
Man. No. 9800611a

f " Introduction to the iRMX 86 Operating System. ", Man. No. 9803124-03.

g " ICE 85b In Circuit Emulator Operating Instructions for ISIS II Users.",
Man. No. 980463-03.

h " ICE 86a In Circuit Emulator Operating Instructions for ISIS II Users.",
Man. No. 9800714-02.

i " Multibus Handbook ", 1983.

j " MCS-86 Macro Assembly Language Reference Manual. ", Man. No.
9800640-02.

k " PLM86 Programming Manual. ", Man. No. 9800466a.

Appendix 1. Notation.

This appendix is intended to clarify the valve and zero crossing numbering scheme used in the thesis. The system used is conventional, but this will remove any possible confusion.

The basic convertor configuration refered to through out this thesis is shown below in figure A1. The standard numbering scheme has been used for the valves, and is as shown in the figure.

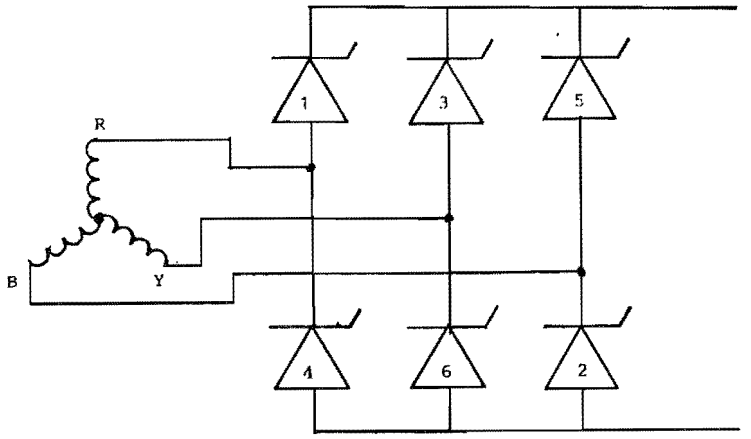


Figure A1 Valve Numbering

The operation of such a bridge is comprehensively explained in many text books, (Kimbark, 1971), (Arrillaga, 1983) and will not be covered in any detail here. The major point of note is that the periods during which any particular valve may conduct are determined by the three phase commutating voltages. These are shown below in figure A2.

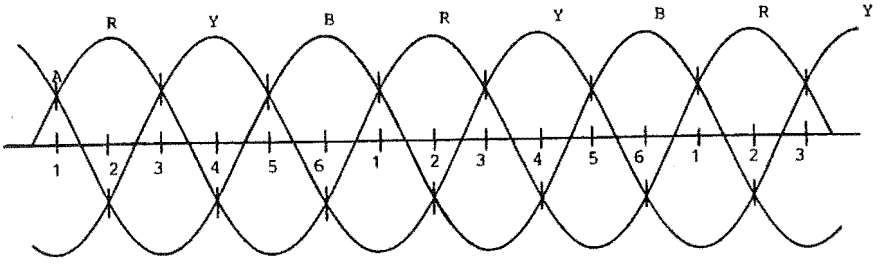


Figure A2 Zero Crossing Numbering

Each valve may conduct only if it is forward biased, although it will not do so unless it is fired. Under normal operating conditions, the three valves in each half of the bridge switch sequentially in numeric order. In the top half of the bridge, valve V1 replaces V5, Valve V3 then replaces V1, valve V5 then replaces V3 and so on. Since each valve has a particular phase - neutral voltage on its cathode, and all three anodes are common, then the voltage that determines whether the particular valve is forward biased is actually a phase - phase voltage. Under normal operating conditions, for valve V1 to begin conduction, the red phase - neutral voltage must be more positive than the blue phase - neutral voltage. This is equivalent to the red - blue phase to phase voltage being positive. The first instant at which valve V1 could start conduction if fired is the instant that this voltage becomes positive. This is shown at point A in figure A2. Since this zero crossing permits valve V1 to start conduction, it is refered to in the thesis as the C1 crossing. The other crossings follow sequentially through the remainder of the cycle, as shown in figure a2.

In the bargraphs, the commutating voltages, VCa, VCb, VCc are the red-blue, yellow-red, and blue-yellow phase - phase voltages respectively. Crossing C1 is therefore the positive going crossing of VCa, crossing C2 is the negative going crossing of VCc, crossing C3 is the positive going crossing of VCb, crossing C4 the negative going crossing of VCa, crossing C5 the positive going crossing of VCc, and crossing C6 the negative going crossing of VCb.

The delay angle of a specific valve is defined as the time from when the valve was first forward biased in the cycle until it is fired. The commutation angle of a valve is the time from when the valve is fired until the valve that was on turns off. The extinction angle of a valve is defined as the time from when the valve turned off until it is next forward biased again.

ERRATTA

P1	3rd line	Replace 'are' with 'is' .
P1	2nd para 7th line	Replace 'Sevenco' with 'Giesbrecht' .
P5		Chapter two heading is omitted .
P9	4th para 3rd line	Replace 'the' with 'The' .
P14	3rd para 4th line	Replace 'mention' with 'mentioned' .
P20	2nd para 2nd line	Replace 'is' with 'in' .
P40	2nd para 7th line	Replace 'the' with 'then' .
P45	2nd para 3rd line	Replace '6.55' with '3.27' .
P47	5th para 2nd line	Replace 'current' with 'voltage' .
P47	5th para 6th line	Replace 'is a' with 'is' .
P59	2nd para 1st line	Replace 'crossing' with 'commutating voltage crossing' .
P72		The captions to fig 6.2 and 6.3 are reversed .
P89	4th para 2nd line	Replace 'in Table 6.1' with 'below' .
P90	1st para 8th line	Replace 'Table 6.2' with 'The table below' .
P101	2nd para 2nd line	Replace 'required' with 'available' .
P105		Left hand figure is 7.4a, right hand 7.4b.
P106	3rd para 1st line	Replace 'indicated' with 'calculated' .
P112	2nd para 3rd line	Replace 'overlap' with 'angle' .
P116	1st para 4th line	Replace 'replace' with 'replaced' .
P124	2nd para 4th line	Replace 'well' with 'well be' .
P125	2nd reference	Replace '1968' with '1967' .